



(Cahiers Mathématiques de l'Université de Sherbrooke)

Revue : CaMUS (Cahiers Mathématiques de l'Université de Sherbrooke)

Volume : 2

Année : 2012

Éditeur : Université de Sherbrooke. Département de Mathématiques

URI : Repéré à : <http://camus.math.usherbrooke.ca/revue.html>

Page vide laissée intentionnellement



Volume 2 Cahiers Mathématiques de
l'Université de Sherbrooke

C. Rousseau <i>Du théorème de Rolle à la théorie de Khovanskii</i>	1
N. Bureau <i>Sensibilité des coups au billard</i>	9
A. Day <i>Interpolation et estimation de dérivées en dimension d selon une approche lagrangienne</i>	27
A. Salvail-Bérard <i>Les arbres de décision hybrides</i>	34
G. Douville <i>Triangulations, carquois et théorème de Ptolémée</i>	59
M. Chabot <i>Fonctions de frises et algèbres amassées</i>	78
M. Toussaint et C. Thibaudeau <i>Analyse et comparaison de procédures d'optimisation en tomodensitométrie</i>	94
F. Camirand <i>Groupe de tresses et les surfaces hyperelliptiques</i>	113
D. Boulet-St-Jacques <i>Les algèbres amassées : définitions de base et résultats</i>	135

F. Baril Boudreau

*Homologie d'une surface de Riemann et problèmes de
Riemann-Hilbert*

151

DU THÉORÈME DE ROLLE À LA THÉORIE DE KHOVANSKII

CHRISTIANE ROUSSEAU

1. Introduction

Beaucoup de problèmes de modélisation se ramènent à solutionner un système d'équations et à compter le nombre de solutions possibles. Des sous-problèmes importants consistent à borner le nombre de solutions d'un système, ou encore à montrer que le nombre de solutions est fini. Dans cet article, notre matériel de base est le théorème de Rolle, et nous montrons comment Askold Khovanskii est parti de questions très simples pour construire une théorie très puissante et élégante : la théorie des « fewnomials ». Michel Rolle est un mathématicien français à cheval sur le 17^e et le 18^e siècle (1652-1719). Askold Khovanskii est un mathématicien russe. Le congrès international se tient tous les 4 ans. Les conférenciers invités sont les mathématiciens dont les travaux de recherche représentent les développements les plus significatifs dans les années précédant le congrès. Pour sa théorie des fewnomials, Askold Khovanskii a été conférencier invité au congrès international des mathématiciens de 1983 à Varsovie.

Rappelons le théorème de Rolle qui est illustré à la figure 1.

THÉORÈME 1.1. *Si une fonction $f : [a, b] \rightarrow \mathbb{R}$ est continue sur $[a, b]$ et dérivable sur $]a, b[$, et si $f(a) = f(b)$, alors il existe $c \in]a, b[$ tel que $f'(c) = 0$.*

EXEMPLE 1.2. *Tout polynôme $P(x)$ à coefficients réels de degré n a au plus n racines réelles. Ceci se prouve facilement par induction. En effet, c'est vrai pour un polynôme de degré 1. Supposons maintenant que ce soit vrai pour tout*

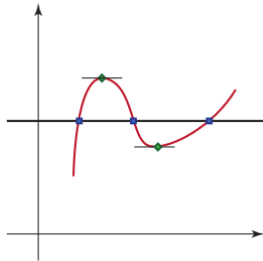


FIGURE 1. Le théorème de Rolle

polynôme de degré n , et soit $P(x)$ un polynôme de degré $n + 1$. Supposons que $P(x)$ ait au moins $n + 2$ racines réelles $x_1 < x_2, \dots < x_{n+2}$. Alors, entre deux racines consécutives x_j and x_{j+1} il existe au moins une racine de la dérivée $P'(x)$. Ceci donne au moins $n + 1$ racines pour P' . Contradiction, puisque P' est de degré n . Cette propriété de finitude du nombre de racines d'un polynôme $P(x)$ vient du caractère algébrique de la fonction $P(x)$. Un fait remarquable est que le théorème de Rolle permet aussi de compter les racines multiples.

Cependant, ce ne sont pas tous les polynômes de degré n qui ont de l'ordre de n racines. En effet, considérons le polynôme $x^n - 1$. Quelle que soit la taille de n , ce polynôme a au plus deux racines réelles.

Pourquoi ?

Réponse

- Parce qu'il a peu de monômes. Un théorème général résumant cette propriété est le théorème de Descartes que l'on décrira ci-dessous.
- Mais Khovanskii est allé plus loin. **Pourquoi un polynôme avec peu de monômes a-t-il peu de racines réelles ?** Nous allons voir sa réponse.

Commençons par une observation qui va nous préparer au théorème de Descartes.

PROPOSITION 1.3. *Un polynôme $P(x) = \sum_{i=1}^m a_i x^{n_i}$, où $n_1 < n_2 < \dots < n_m$ et $a_i \neq 0$ pour tout i , a au plus $m - 1$ racines réelles positives et au plus $m - 1$ racines réelles négatives, en comptant les multiplicités. Il a donc au plus $2(m - 1) + n_1$ racines réelles en comptant les multiplicités.*

DÉMONSTRATION. La preuve se fait par induction sur m . Si $m = 1$, on a un seul monôme qui ne s'annule qu'en $x = 0$ avec multiplicité n_1 . Supposons que la proposition soit vraie pour un polynôme ayant m monômes, et considérons $P(x) = \sum_{i=1}^{m+1} a_i x^{n_i}$. Il suffit de s'occuper des racines positives et de remarquer que les racines négatives sont les racines positives du polynôme $P(-x)$. Alors x est une racine positive de $P(x)$ si et seulement si x est une racine positive de $Q(x) = \frac{P(x)}{x^{n_1}}$. On a $Q(x) = a_1 + \sum_{i=2}^{m+1} a_i x^{s_i}$, où $s_i = n_i - n_1 > 0$. Entre deux racines positives de $Q(x)$ il existe une racine positive de $Q'(x)$. Mais $Q'(x) = \sum_{i=2}^{m+1} s_i a_i x^{s_i-1}$ a au plus $m - 1$ racines positives par l'hypothèse d'induction. Donc, $Q(x)$ a au plus m racines positives.

Déjà nous voyons une généralisation importante : dans la preuve de la proposition, lorsque nous avons déterminé une borne supérieure pour le nombre de racines positives, nous n'avons utilisé à aucun moment le fait que les n_i sont des entiers. La proposition reste vraie avec la même preuve si les n_i sont des nombres réels ! C'est le corollaire

COROLLAIRE 1.4. *Une fonction $f(x) = \sum_{i=1}^m a_i x^{\alpha_i}$, où $\alpha_1 < \alpha_2 < \dots < \alpha_m$ et $a_i \neq 0$ pour tout i , $\alpha_i \in \mathbb{R}$, a au plus $m - 1$ racines réelles positives en comptant les multiplicités.*

La fonction $f(x)$ du corollaire 1.4 n'est plus algébrique sur $]0, \infty[$, mais seulement analytique. Pourtant, elle a, comme un polynôme, un nombre fini de racines.

Pourquoi ?

Nous y reviendrons. Mais nous allons commencer par raffiner la proposition 1.3 pour aboutir au théorème de Descartes. En effet, pour qu'un polynôme $P(x)$ ait des racines positives, il faut que tous ses coefficients ne soient pas de même signe.

THÉORÈME 1.5. (*théorème de Descartes*) Soit $P(x) = \sum_{i=1}^m a_i x^{n_i}$, où $n_1 < n_2 < \dots < n_m$ et $a_i \neq 0$ pour tout i , un polynôme, et soit r le nombre de changements de signe dans la suite des coefficients a_1, \dots, a_m . Alors $P(x)$ a au plus r racines réelles positives en comptant les multiplicités.

DÉMONSTRATION. La démonstration suivante utilisant le théorème de Rolle a été donnée par Laguerre. Elle se fait par induction sur r . Le théorème est vrai pour $r = 0$. Supposons qu'il soit aussi vrai si on a r changements de signe et considérons un polynôme $P(x)$ avec $r+1$ changements de signe. Nous allons écrire $P(x) = \sum_{j=0}^{r+1} p_j(x)$, où $p_j(x)$ est une somme de monômes dont les coefficients sont de même signe, et les coefficients de p_j et p_{j+1} sont de signe opposé. Il est clair que x^α n'a pas de racine réelle positive pour tout nombre réel non nul α . Donc, le nombre de racines réelles positives de $P(x)$ est le même que le nombre de racines positives de $Q(x) = \frac{P(x)}{x^\alpha}$. Par le théorème de Rolle, ce nombre est au plus 1 plus le nombre de racines positives de $Q'(x)$. On a que

$$Q'(x) = \frac{\sum_{j=0}^{r+1} q_j(x)}{x^{\alpha+1}},$$

où $q_j(x) = p'_j(x)x - \alpha p_j(x)$. Le nombre de racines positives de $Q'(x)$ est bien sûr celui de $R(x) = \sum_{j=0}^{r+1} q_j(x)$. Pour montrer le théorème, il suffit de voir qu'on peut bien choisir le α de sorte que $R(x)$ n'ait que r changements de signe, et donc, au plus r racines. Regardons maintenant chacun des q_j . Supposons que $p_j(x) = \sum_{i=1}^{s_j} b_{ij} x^{k_{ij}}$ où $k_{1j} < k_{2j} < \dots < k_{s_j j}$. Alors $q_j(x) = \sum_{i=1}^{s_j} b_{ij} (k_{ij} - \alpha) x^{k_{ij}}$. Il suffit donc de choisir α tel que, par exemple, $k_{s_0 0} < \alpha < k_{11}$, c'est-à-dire que α est compris entre le plus grand exposant de p_0 et le plus petit exposant de p_1 . Alors, le signe des coefficients des q_j est le même que celui des coefficients des p_j pour $j \geq 1$, et l'inverse du signe des coefficients de p_0 quand $j = 0$. Donc, on a éliminé un changement de signe en passant de $P(x)$ à $Q(x)$, soit le changement de signe entre p_0 et p_1 . Ainsi, de par l'hypothèse d'induction, $R(x)$ a au plus r changements de signe, et donc, r racines. Par suite, $P(x)$ a au plus $r+1$ racines.

Encore une fois, nous aurions pu remarquer que le théorème serait resté vrai si les n_i avaient été des nombres réels plutôt que des entiers.

Ces théorèmes se généralisent aux systèmes de n polynômes à n variables, mais nous nous contenterons de citer les résultats sans les prouver.

THÉORÈME 1.6. (*théorème de Bezout*) Si $P(x, y)$ et $Q(x, y)$ sont deux polynômes à coefficients réels de degrés respectifs m et n , alors le système d'équations

$$\begin{cases} P(x, y) = 0, \\ Q(x, y) = 0, \end{cases}$$

a au plus mn solutions (x_i, y_i) dans \mathbb{R}^2 en comptant les multiplicités.

Il y a moyen de donner une preuve de ce théorème en utilisant la théorie de l'élimination et en se ramenant au cas d'un polynôme à un variable. Ce qui nous intéresse plutôt, ce sont les généralisations qu'en a données Khovanskii dans le cas de systèmes où $P(x, y)$ et $Q(x, y)$ ont un petit nombre de monômes.

THÉORÈME 1.7. (*Khovanskii*) *Le nombre de solutions réelles non dégénérées d'un système de n équations polynomiales à n inconnues ayant q monômes est inférieur ou égal à*

$$2^{q(q-1)/2}(n+1)^q.$$

Nous avons laissé en suspens deux grandes questions auxquelles Khovanskii a donné une réponse.

- (1) Pourquoi les polynômes ou systèmes polynomiaux ayant peu de monômes ont peu de solutions réelles ?
- (2) Pourquoi certaines fonctions analytiques ont des propriétés de finitude comme les fonctions algébriques ?

Ces questions ne sont que des cas particuliers de la grande théorie créée par Khovanskii qui continue, depuis plus de 25 ans, d'être raffinée pour solutionner des problèmes de recherche de plus en plus complexes.

2. La théorie des fewnomials

La théorie des fewnomials explique pourquoi les polynômes ou systèmes polynomiaux ayant peu de monômes ont peu de solutions réelles. En effet, revenons sur l'exemple du polynôme $P(x) = x^n - 1$. Ce polynôme a n racines complexes qui sont les n racines de l'unité. Ces racines ont

- même module,
- des arguments uniformément répartis dans $[0, 2\pi]$.

La première propriété ne se généralisera pas pour un polynôme quelconque ayant un petit nombre de monômes, mais la deuxième restera valide sous une forme plus faible. Voici le théorème montré par Khovanskii. Il est un peu technique, mais il faut faire l'effort d'en comprendre l'idée.

THÉORÈME 2.1. *On considère un système $P(x_1, \dots, x_n) = 0$, où $P = (P_1, \dots, P_n)$ et les $P_j(x_1, \dots, x_n)$ sont des polynômes dans $\mathbb{R}[x_1, \dots, x_n]$. Soit*

$$\mathbb{T}^n = \underbrace{\mathbb{S}^1 \times \dots \times \mathbb{S}^1}_n$$

le tore de dimension n qui est donné par le produit cartésien de n cercles. Sur \mathbb{T}^n on considère des coordonnées angulaires $(\theta_1, \dots, \theta_n)$ modulo 2π . Soit Ω une région de \mathbb{T}^n . À chaque solution de composantes non nulles $x = (x_1, \dots, x_n) \in \mathbb{C}^n$ de $P(x) = 0$, on associe le vecteur $\theta = (\theta_1, \dots, \theta_n)$ où $\theta_j = \arg x_j$. Soit $N(P, \Omega)$ le nombre de θ dans Ω . Alors, il existe des constantes $\Pi = \Pi(\Omega, \text{Newton})$ et $\phi(n, g)$, où Π dépend de Ω et de l'enveloppe convexe des exposants des monômes du système (aussi appelé polygone de Newton), et ϕ dépend du nombre de monômes g

du système $P(x) = 0$ et de la dimension n telles que

$$|N(P, \Omega) - N(P, \mathbb{T}^n)| \frac{\text{vol}(\Omega)}{\text{vol}(\mathbb{T}^n)} \leq \Pi \phi(g, n).$$

En pratique, le produit $\Pi \phi(g, n)$ est petit lorsque le système a peu de monômes, ce qui signifie que le quotient $\frac{N(P, \Omega)}{N(P, \mathbb{T}^n)}$ est assez proche de $\frac{\text{vol}(\Omega)}{\text{vol}(\mathbb{T}^n)}$. Le concept de polygone de Newton est un concept technique qui ne sera pas décrit ici.

3. Les fonctions de Pfaff

Il existe une classe de fonctions analytiques qui ont de bonnes propriétés de finitude comme les fonctions algébriques. Ce sont les *fonctions de Pfaff*. La raison en est que ces fonctions ont une origine algébrique. Dans le cas des fonctions à une variable, ce sont des solutions d'équations différentielles algébriques :

$$(1) \quad \frac{dy}{dx} = \frac{P(x, y)}{Q(x, y)},$$

où $P(x, y)$ et $Q(x, y)$ sont des polynômes à coefficients réels. (En fait, toutes les solutions d'un tel système ne seront pas des fonctions de Pfaff, mais seulement les solutions *séparantes*. Nous discuterons cette distinction plus bas.)

EXEMPLE 3.1. (1) La fonction $y = x^\alpha$, où $\alpha \in \mathbb{R}$ est une fonction de Pfaff. En effet, elle est solution de l'équation différentielle

$$\frac{dy}{dx} = \frac{\alpha y}{x}.$$

Rappel : ce sont des fonctions de cette forme qui ont été utilisées dans le corollaire 1.4.

(2) La fonction $y = e^{ax}$ est une fonction de Pfaff. En effet, elle est solution de l'équation différentielle

$$\frac{dy}{dx} = ay.$$

(3) La fonction $y = x \ln x$ est une fonction de Pfaff. En effet, elle est solution de l'équation différentielle

$$\frac{dy}{dx} = \frac{y}{x} + 1.$$

L'avantage de cette approche est que, lorsque nous considérons une équation différentielle, nous n'avons pas une fonction de Pfaff, mais toute une famille de fonctions de Pfaff qui sont solutions de la même équation différentielle. Cela a permis à Khovanskii de généraliser le théorème de Rolle sous la forme suivante.

THÉORÈME 3.2. (théorème de Rolle pour les systèmes dynamiques). On considère une région du plan, Ω , remplie de solutions d'une équation différentielle de la forme (1), et une solution particulière (séparante) γ (voir figure 2). Soit \mathcal{C} une courbe de classe C^1 dans Ω . Alors, entre deux points d'intersection consécutifs de γ et de \mathcal{C} il existe sur \mathcal{C} un point où \mathcal{C} est tangente à une des solutions de l'équation différentielle (1) (un tel point est appelé un point de contact).

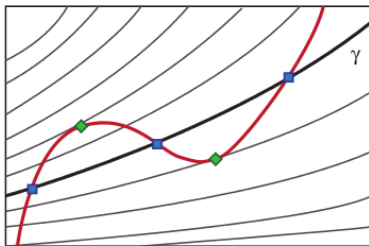
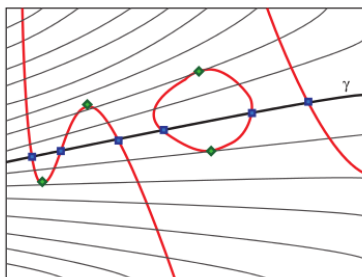


FIGURE 2. Le théorème de Rolle pour les systèmes dynamiques

FIGURE 3. Points d'intersection et points de contact d'une solution séparante γ avec une courbe algébrique (en rouge)

La figure 2 illustre parfaitement la signification du théorème. Le théorème de Rolle que nous connaissons est un cas particulier de ce théorème. Il suffit de regarder l'équation différentielle $\frac{dy}{dx} = 0$, dont les solutions sont données par la famille des droites horizontales $y = C$.

DÉFINITION 3.3. Une courbe algébrique de degré n est l'ensemble des points (x, y) du plan tels que $F(x, y) = 0$, où $F \in \mathbb{R}[x, y]$ est un polynôme en x, y à coefficients réels.

COROLLAIRE 3.4. Le nombre de points d'intersection isolés d'une courbe algébrique $F(x, y) = 0$ de degré m avec une solution séparante d'un système différentiel de la forme (1) de degré n est au plus $m(n+m)$. (Le degré d'une équation de Pfaff (1) est le maximum des degrés de P et Q .)

DÉMONSTRATION. Sur chaque composante fermée, le nombre de points d'intersection est inférieur ou égal au nombre de points de contact. Sur chaque composante infinie, on a au moins un point de contact entre deux points d'intersection. Donc, le nombre de points d'intersection est inférieur ou égal au nombre de points de contact plus le nombre de composantes infinies (figure 3). Le nombre de points de contact est le nombre de solutions du système

$$\begin{cases} F_x Q + F_y P = 0, \\ F = 0, \end{cases}$$

puisque $F_x P + F_y Q = 0$ exprime que $\nabla F = (F_x, F_y)$ est perpendiculaire au champ de vecteurs $(P(x, y), Q(x, y))$. La première équation est polynomiale de degré $m - 1 + n$ et la deuxième de degré m . Donc, par le théorème de Bezout, le nombre de solutions est inférieur ou égal à $m(m + n - 1)$.

Il faut maintenant borner le nombre de composantes infinies. Les composantes infinies intersectent tout cercle $x^2 + y^2 = R$ en au moins deux points. Donc, le nombre de composantes infinies est inférieur ou égal à la moitié du nombre de points d'intersection de

$$\begin{cases} x^2 + y^2 = R, \\ F(x, y) = 0, \end{cases}$$

soit $\frac{1}{2}2m = m$. Le nombre de points d'intersection est donc inférieur ou égal à

$$m(m + n - 1) + m = m(m + n).$$

Regardons le miracle qui s'est produit : on a transformé la recherche du nombre de solutions d'un problème analytique en la recherche du nombre de solutions de deux problèmes algébriques. Donc, on a mis en évidence le caractère algébrique caché qui explique la finitude du nombre de solutions.

DÉFINITION 3.5. *On considère une équation de Pfaff (1), dont on oriente les trajectoires selon la direction du champ de vecteurs tangents $(P(x, y), Q(x, y))$. Une solution (ou un ensemble fini de solutions), γ , est une solution séparante de l'équation de Pfaff si γ est la frontière orientée d'un domaine dans \mathbb{R}^2 .*

La figure 4 donne des exemples de solutions non séparantes et séparantes. On voit que la condition que la solution est séparante est exactement ce qu'il faut pour que le théorème 3.2 fonctionne.

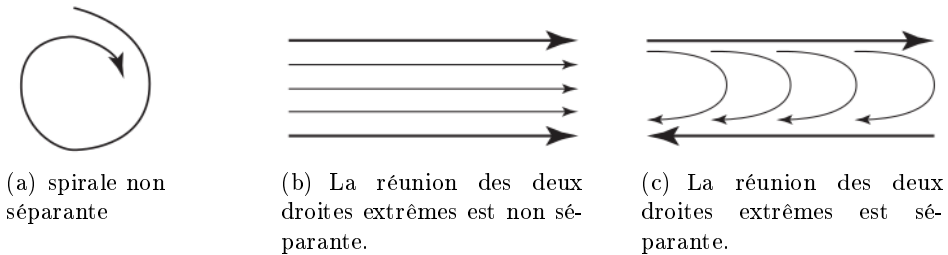


FIGURE 4. Exemples de solutions non séparantes et séparantes

Ce théorème se généralise à des systèmes dans \mathbb{R}^n qui sont un mélange d'équations algébriques et d'équations pfaffiennes. De nombreuses variantes de la méthode existent que nous ne discuterons pas ici. Et les chercheurs qui utilisent la méthode créent les nouveaux raffinements que leurs systèmes d'équations exigent. Nous nous contenterons, pour terminer, de donner un exemple de théorème montré par Khovanskii.

THÉORÈME 3.6. *Soit $a_1, \dots, a_k \in \mathbb{R}^n$. On considère un ensemble d'équations $P_1(x_1, \dots, x_n) = \dots = P_n(x_1, \dots, x_n) = 0$, où P_j est un polynôme de degré d_j*

dans les variables x_i et les fonctions $\exp(\langle a_s, x \rangle)$. Alors, le nombre de racines non dégénérées de ce système dans \mathbb{R}^n est au plus

$$2^{\frac{k(k+1)}{2}} \left(\sum_{j=1}^n (d_j + 1) \right)^k d_1 \dots d_n.$$

4. Conclusion

Ce petit article veut illustrer qu'il n'y a pas de questions trop simples pour le scientifique à condition de garder les yeux ouverts, et qu'il n'est pas nécessaire d'avoir des connaissances très avancées pour faire des percées scientifiques significatives.

Références

- [Kho82] A. Khovanskii, Cycles of dynamical systems on the plane and Rolle's theorem, *Funct. Anal. Appl.*, Sibirskii Math. Journal, **25** (1984), 502–506 (version originale en russe publiée en 1982).
- [Kho] A. Khovanskii, Theory of fewnomials.
- [Kho83] A. Khovanskii, Fewnomials and Pfaff manifolds, Proceedings of the International Congress of Mathematicians, 1983, Warszawa, 549–564.

CHRISTIANE ROUSSEAU, DÉPARTEMENT DE MATHÉMATIQUES, UNIVERSITÉ DE MONTRÉAL
Courriel: rousseac@DMS.UMontreal.CA

SENSIBILITÉ DES COUPS AU BILLARD

NICOLAS BUREAU

RÉSUMÉ. Le but de cette analyse est de quantifier la sensibilité d'un coup au billard. Pour une poche et une position de bille objet données, il existe une marge d'erreur, un angle, qui permet de réussir un coup. Cette quantité est très importante, car elle aide ensuite à déterminer une deuxième marge d'erreur : celle qui relie la bille blanche à la bille que l'on veut empocher. Cet angle est un reflet direct de la difficulté et de la sensibilité d'un coup. Ultimement, cette quantité est le cœur de cette recherche. Pour atteindre ce but, la géométrie vectorielle est l'outil d'excellence.

1. Introduction

Le billard est un jeu d'adresse, de précision et de stratégie. Débutant ou professionnel, lorsqu'il frappe la blanche dans le but d'empocher une autre bille, le joueur risque de perdre son tour à la moindre erreur. Pourquoi en est-il ainsi ? Peut-on déterminer une certaine marge d'erreur admissible qui permet de réussir un coup ? A priori, il est difficile de répondre à cette question, parce que beaucoup de facteurs peuvent amplifier les erreurs. De plus, la modélisation mathématique et physique d'un jeu de billard est une tâche complexe en soi. Un joueur de billard virtuel, comme celui de mon collègue Jean-François Landry [LDM11], a besoin de comprendre ce phénomène afin de minimiser ses erreurs et de maximiser ses chances de gagner un tournoi. Néanmoins, on peut toujours étudier un cas simple avec lequel on est particulièrement à l'aise.

L'objectif de cette étude est de considérer des coups de billard linéaires pour en analyser la sensibilité, c'est-à-dire la marge d'erreur. Certes, un coup linéaire simplifie de beaucoup la problématique, mais il permet quand même de dessiner un bon portrait d'un type de coup très commun au billard. De ce fait, l'aspect physique est mis de côté dans le but de faire une étude plutôt géométrique et quelque peu vectorielle.

L'idée est de commencer avec une poche de coin ou de bord et d'en extraire le plus d'information possible. Puis on y ajoute une bille et on détermine les différentes méthodes qui permettent d'empocher la bille avec la meilleure marge d'erreur possible. Bien entendu, cette erreur est quantifiable et donc calculable.

Le support majeur de Jean-Pierre Dussault ainsi que ses conseils judicieux ont été grandement appréciés. Merci également au CRSNG et, une fois de plus, à Jean-Pierre Dussault pour leur soutien financier pour ce stage en recherche.

Ensuite, on ajoute la blanche et on fait une analyse similaire : en fonction de la configuration des billes données, comment peut-on déterminer une marge d'erreur qui va permettre d'empocher la bille visée ? Toutes ces questions seront étudiées en temps et lieu, mais avant toute chose, on doit se fixer un modèle mathématique de la table.

2. Préliminaires

Il existe plusieurs types de tables de billard. Ici, on s'intéresse uniquement à une table de billard américain modélisée de la manière suivante :

- le coin supérieur droit est le point $(0,0)$ du plan cartésien ;
- les abscisses sont positives à gauche de l'origine et les ordonnées sont positives sous l'origine ;
- la longueur de la table est de L (sur les abscisses) et sa largeur est de H (sur les ordonnées) ;
- la poche supérieure droite (la première étudiée) est délimitée par les coordonnées $(h,0)$ et $(0,h)$;
- la poche supérieure centrale (la deuxième étudiée) est délimitée par les coordonnées $(m,0)$ et $(0,m)$;
- une bille a un rayon de r ;
- la position d'une bille est déterminée par son centre ;
- Les portions de la bande (AB , BC et CD), illustrées dans la Figure 1, se nomment *bandes intérieures des poches*.

De plus, pour rendre les équations plus lisibles, on utilisera souvent les constantes α , β , γ et δ dans ce document. La variable α représente l'angle entre le bord de la table et l'ouverture de la poche de coin, tandis que β est l'angle entre le bord et l'ouverture de la poche centrale.

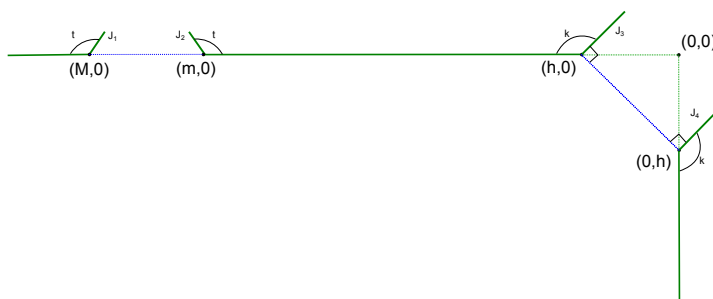


FIGURE 1. Poches de côté et de coin

3. Sensibilité de la balle objet

Cette section a pour but d'analyser la sensibilité de la balle objet à empocher. À partir d'une position donnée de la bille en question, on doit déterminer l'angle maximal pour empocher celle-ci. On commencera avec le cas d'une poche de coin, puis viendra l'analyse de la poche de côté.

3.1. Poche supérieure droite

L'analyse se fait en trois temps, car il existe trois régions de position de la bille qui nécessitent des équations différentes. Le diagramme suivant illustre ces emplacements :

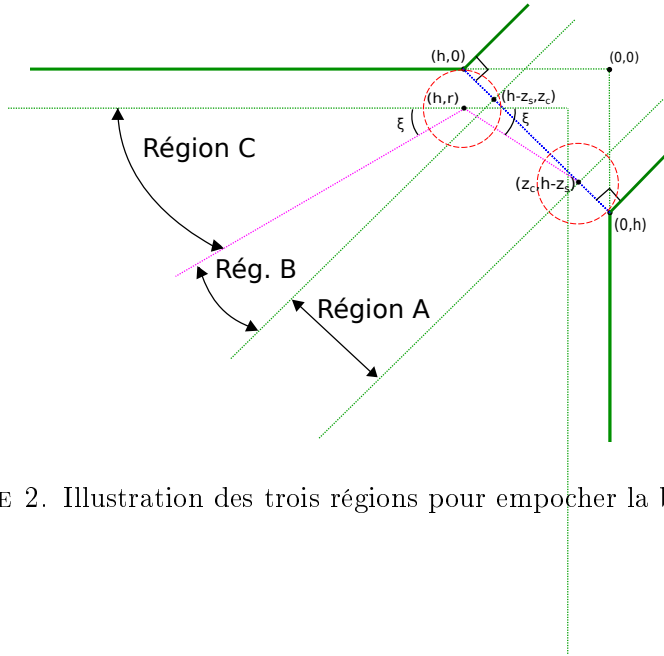


FIGURE 2. Illustration des trois régions pour empocher la balle objet

Seulement la moitié supérieure des régions est illustrée (et seulement cette moitié sera présentée), mais il est facile de s'imaginer, de façon symétrique, les régions pour la partie inférieure du dessin. La région A est celle dans laquelle une bille peut atteindre directement les deux bandes de l'intérieur de la poche. La région B est celle dans laquelle une bille peut atteindre une seule des bandes intérieures, mais pas l'autre. Ainsi, la trajectoire de la balle objet doit être tangente au coin de la poche. Finalement, la région C est celle qui permet l'utilisation de la bande pour améliorer la marge d'erreur.

Région A : Coup direct. Le coup direct est le cas le plus facile à calculer. Il faut d'abord trouver les positions limites atteignables près de la poche.

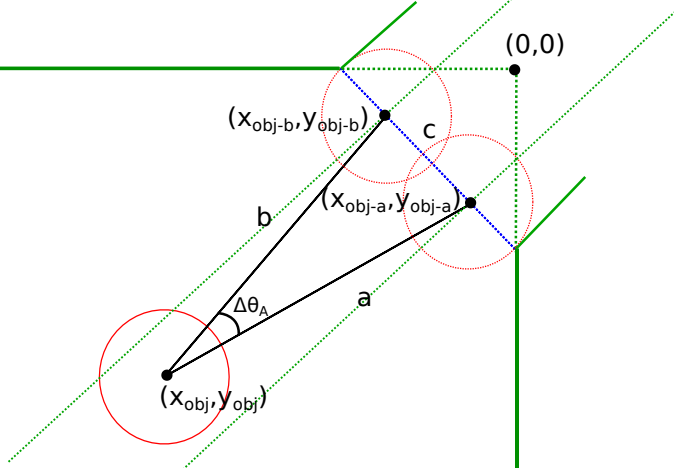


FIGURE 3. Illustration de la région A

On peut voir, dans la figure ci-haut, les bandes de la table en vert, la délimitation de la poche en bleu et les positions limites de la bille objet en rouge pointillé. Ces billes (dites fantômes) sont centrées à exactement un rayon des deux coins de l'intérieur de la poche. Ainsi, une distance de $c = \sqrt{2} \cdot (h - z_c - z_s)$ sépare ces deux billes.

Définissons le segment le plus à droite a et celui le plus à gauche b . La position de la bille objet est (x_{obj}, y_{obj}) . La position des billes fantômes sont respectivement $(x_{obj-b}, y_{obj-b}) = (h - z_s, z_c)$ et $(x_{obj-a}, y_{obj-a}) = (z_c, h - z_s)$. Ainsi, on peut calculer les valeurs suivantes :

$$b = \sqrt{(h - z_s - x_{obj})^2 + (z_c - y_{obj})^2}$$

$$a = \sqrt{(h - z_s - y_{obj})^2 + (z_c - x_{obj})^2}$$

Finalement, l'angle créé par l'intersection de a et b mesure :

$$\Delta\theta_A = \arccos\left(\frac{a^2 + b^2 - c^2}{2ab}\right)$$

La région A est délimitée pour deux droites imaginaires (d'une distance d'un rayon de bille) à la bande de l'intérieur de la poche. Ainsi cette région est définie par :

$$\begin{aligned} y &\geq \tan(180 - k) \cdot (x - h) + r \cdot \sec(180 - k) \\ y &\leq \cot(180 - k) \cdot x - r \cdot \csc(180 - k) + h \end{aligned}$$

Région C : Utilisation de la bande. Plus l'angle $\Delta\theta$ est grand, meilleures seront les chances de réussir un coup, il serait donc avantageux de trouver un moyen de l'améliorer. L'utilisation des bandes pour faire des rebonds pourrait aider à atteindre ce but. Toutefois, pour des raisons de simplification, on choisit pour hypothèse que, lors d'un rebond sur la bande, l'angle d'incidence correspond exactement à l'angle de réflexion.

Avant toute chose, il faudrait trouver une formule qui détermine la position à atteindre sur la bande pour empêcher la balle objet. L'image suivante montre l'emplacement pour empêcher la balle objet en utilisant la bande supérieure.

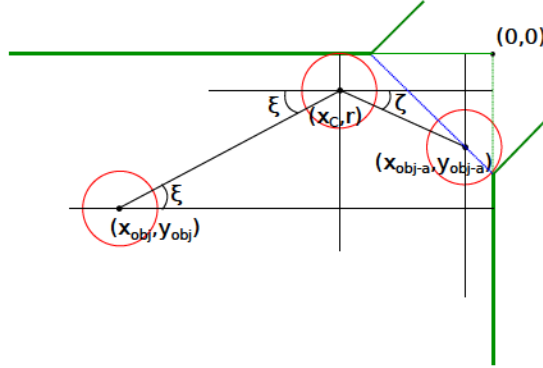


FIGURE 4. Empêcher la balle objet en utilisant la bande du haut

On déduit de ci-haut l'équation $\tan \zeta = \frac{y_{obj-a} - r}{x_C - x_{obj-a}} = \frac{h - z_s - r}{x_C - z_c}$, où x_C est la position recherchée sur l'axe des abscisses où la balle objet entre en collision avec la bande. On déduit également l'équation $\tan \xi = \frac{y_{obj} - r}{x_{obj} - x_C}$. Comme l'angle d'incidence est égal à l'angle de réflexion, on obtient l'identité $\frac{y_{obj} - r}{x_{obj} - x_C} = \frac{h - z_s - r}{x_C - z_c}$, d'où on obtient la formule suivante :

$$x_C = \frac{x_{obj}(h - z_s - r) + z_c(y_{obj} - r)}{y_{obj} - 2r + h - z_s}$$

Une fois cette quantité trouvée, on peut s'intéresser au nouvel angle $\Delta\theta_C$ créé lors de l'utilisation de la bande. Il s'agit en fait de l'angle illustré ci-dessous :

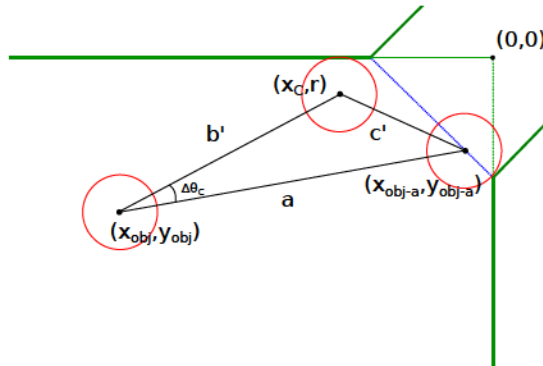
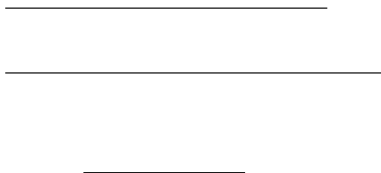


FIGURE 5. Nouvel angle créé par l'utilisation de la bande supérieure

La longueur l est celle que l'on retrouve dans la Figure 3. Il reste à déterminer les côtés a et b du triangle, puis l'angle θ :



Maintenant que l'on sait trouver la position de la collision avec les bandes et que l'on peut déterminer l'angle produit par cette méthode, on peut se poser la question : à quel moment est-il plus avantageux d'utiliser la bande ? Comme l'illustre la Figure 2, l'emplacement limite sur la bande pour lequel on peut faire un rebond dans le but d'empocher la bille se trouve à $l/2$. Sachant que le rebond se fait exactement à cet endroit, on veut également empocher la bille en faisant un deuxième rebond sur la bande de l'intérieur de la poche. On peut facilement vérifier que la position limite qui permet de le faire se trouve en $l/4$.

Pour trouver la droite qui délimite la région C, il faut trouver deux points qui passent par celle-ci. On connaît déjà le point $(l/4, 0)$ et comme on tient pour acquis que l'angle de réflexion est égal à l'angle d'incidence, ceci veut dire que le graphique est symétrique en $y = l/4$, donc la droite passe par le point $(l/4, l/4)$.

On peut ainsi trouver l'équation $y = x$. La région C est délimitée par la relation :



Région B : Tangente au coin. Selon les limites des deux autres régions, on déduit que la région B est délimitée par la relation suivante :



Dans cette région, l'angle pour réussir le coup sera maximal si la balle frôle le coin supérieur de la poche, comme dans l'illustration suivante :

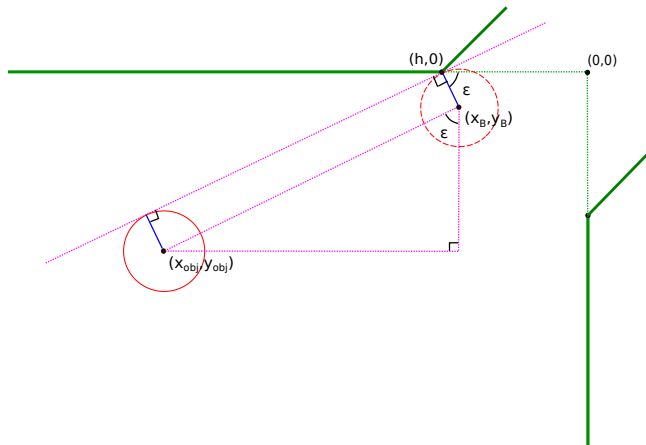


FIGURE 6. Direction pour que la balle objet frôle le coin supérieur de la poche

À première vue, on aurait tendance à vouloir déterminer les coordonnées de la balle fantôme qui effleure le coin supérieur en $(0,0)$, mais on peut se contenter de trouver (x_B, y_B) , l'angle entre le rayon de la balle fantôme, qui est perpendiculaire à la tangente au coin, et l'axe des abscisses vers l'origine. On peut facilement déduire que cet angle est le même que l'angle supérieur du triangle formé par la balle objet et son fantôme, comme dans la figure précédente. On peut ainsi trouver les deux équations suivantes :

$$\begin{aligned} & \text{---} \\ & \text{---} \end{aligned}$$

Or on peut aussi exprimer la coordonnée y_B en fonction de l'angle θ à partir de la position (x_B, y_B) :

En comparant les deux autres équations précédentes et en utilisant cette dernière substitution, on trouve :

En utilisant la substitution $y_B = x_B \tan(\theta)$, $x_B = y_B \cot(\theta)$ et $y_B = x_B \tan(\theta)$, on peut isoler l'angle :

$$\begin{aligned} & \text{---} \\ & \text{---} \end{aligned}$$

Si l'on veut connaître la position de la balle fantôme au point $(0,0)$, il suffit de remplacer cet θ dans $y_B = x_B \tan(\theta)$. Il ne manque plus qu'à déterminer l'angle θ qui permet d'empocher la balle objet :

rebondit sur la bande intérieure. Dans le cadre de cette analyse, on va uniquement considérer les billes qui franchissent l'entrée de la poche. Tout comme pour les poches de coin, trois régions existent pour lesquelles des calculs différents doivent être effectués :

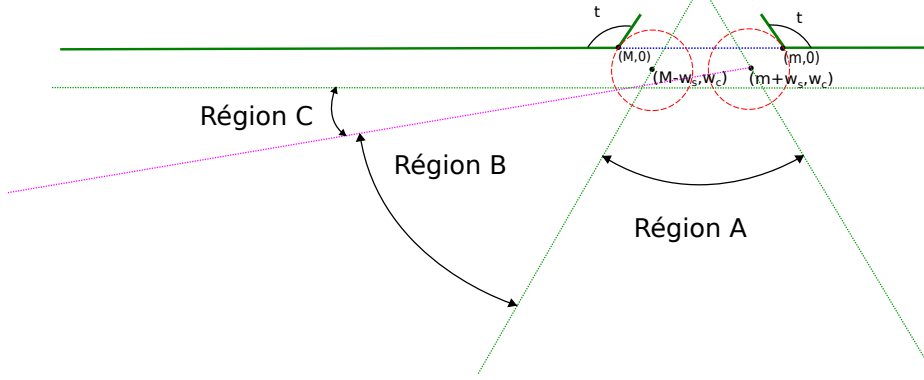


FIGURE 8. Illustration des trois régions autour d'une poche de côté

La région A correspond à l'endroit à partir duquel on peut atteindre les deux billes fantômes directement. La région B correspond à l'endroit à partir duquel on peut atteindre la deuxième bille fantôme, mais pas la première, parce que le coin de la poche l'empêche. Finalement, la région C correspond à l'endroit à partir duquel on ne peut atteindre aucune des deux billes fantômes et donc, on ne peut empocher de bille de façon directe. Il est à noter que les mêmes régions sont présentes à droite de la poche mais, les calculs étant symétriques, on ne les considère pas.

Région A : Coup direct. Dans la région A, la bille objet peut atteindre ses deux fantômes directement.

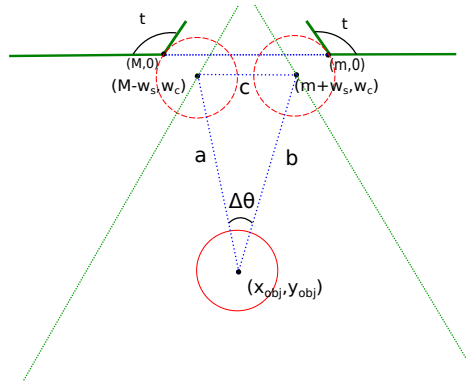


FIGURE 9. Bille objet dans la région A

Il s'agit exactement des mêmes calculs que l'on a effectués dans la Section 3.1 :

Afin de trouver , on détermine la distance entre la bille objet et ses fantômes, puis la distance entre les deux fantômes :

Pour déterminer la limite de la région, on utilise la même technique. L'une des deux droites est nécessairement celle que l'on a trouvée pour la région A : , l'autre est illustrée ci-bas (fuchsia) :

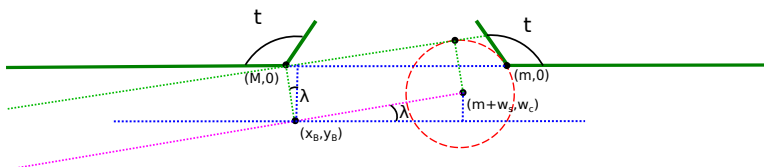


FIGURE 11. Illustration de la limite de la région B (droite fuchsia)

La coordonnée est un point de la droite recherchée. On peut l'exprimer en fonction de , l'angle illustré.

De plus, on observe une autre identité avec ce même angle :

On peut ainsi trouver :

On peut isoler en posant —, — et — :

Comme on sait que la droite limite passe par les points
et , on trouve aisément l'équation :

On a donc les deux inéquations qui délimitent la région B :

Région C : Zone impossible. En regardant la Figure 8, on voit que si une bille se trouve dans la région C, alors elle ne peut atteindre ni l'une et ni l'autre des fantômes, même avec une direction qui coupe le coin de la poche. La seule façon d'empocher une telle bille serait de faire un rebond sur une autre bande (typiquement la bande opposée) ou sur une autre bille, mais ce problème ne sera pas étudiée dans le cadre de cette analyse.

Bien entendu, on a déjà déterminé l'équation qui délimite cette région dans la Section 3.2 :

<i>Résumé.</i>	_____	avec	,	et
Coup direct				
Tangente au coin				

On définit _____, le vecteur pointant vers la direction du centre de l'angle _____, à partir du centre de la bille objet.

4. La blanche

Cette section analyse la sensibilité d'un coup sur la blanche dans le but d'empocher une bille objet visée. Ultimement, le but est de déterminer l'angle _____, dit admissible, permettant de réussir un coup, en fonction de la position de la blanche, de la position de la bille objet et en fonction de l'angle maximal pour empocher cette dernière. Ce dernier angle correspond à _____, calculé dans la section précédente, qui est en fonction de la position de la balle objet (et de la région dans laquelle elle se situe). En premier, on s'intéresse à un coup direct sans contrainte. En deuxième, on étudie un coup très coupé, puis on analyse les contraintes de la bande. Finalement, on analyse rapidement un coup de rebond

sur la bande, en supposant bien sûr que l'angle d'incidence correspond à l'angle de réflexion. Dans cette section, on choisit pour hypothèse que les collisions entre les billes ne sont pas inélastiques et qu'elles n'induisent aucun frottement.

4.1. Coup direct

Ce coup est le plus simple : l'angle θ forme un arc de cercle sur la bille et on peut directement atteindre tout point de cet arc à partir de la position de la blanche :

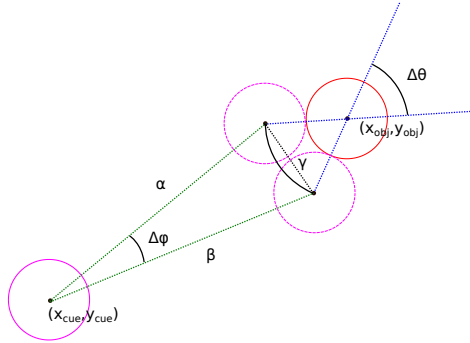


FIGURE 12. Disposition d'un coup direct

Le cercle rouge représente la bille objet en (x_{obj}, y_{obj}) le cercle rose représente la blanche en (x_{cue}, y_{cue}) et les cercles pointillés correspondent aux endroits limites atteignables par la blanche pour réussir le coup (blanches fantômes). Les distances d_{cue} et d_{obj} se déterminent en fonction de la position de la blanche et de l'emplacement de ses fantômes. Les positions de ces billes imaginaires se trouvent toutes deux à une distance de r de la bille objet sur le long des vecteurs qui forment l'angle θ . Ces vecteurs sont facilement déterminables sachant que, dans la section précédente, on a pu trouver la position des fantômes de la bille objet, peu importe la position (x_{cue}, y_{cue}) . Ainsi, dans le reste de cette section, on présuppose que l'on connaît les distances d_{cue} et d_{obj} .

L'angle recherché se trouve donc directement :



4.2. Angle coupé

Il arrive parfois que l'arc de cercle produit par l'angle θ ne soit pas totalement accessible par la blanche.

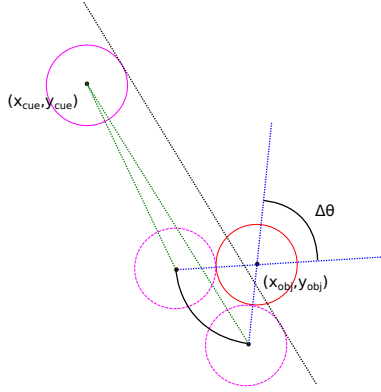


FIGURE 13. Arc de cercle qui ne peut pas être totalement accessible

La figure ci-haut montre que la fantôme du bas n'est physiquement pas atteignable par un coup linéaire de la blanche. En effet, la droite pointillée noire passe à travers la bille objet. Il faut donc déterminer l'endroit limite qui est accessible par l'intermédiaire de la blanche. Cette position est celle qui fait en sorte que la droite pointillée noire soit tangente à la blanche et à la bille objet :

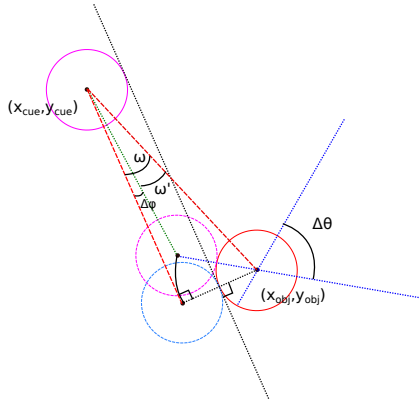


FIGURE 14. Correction de l'angle de tir

Dans cette figure, la balle bleue correspond à la position corrigée de la fantôme blanche. Cet emplacement est le meilleur possible pour que le coup soit exécutable. Pour déterminer ω , on n'a qu'à déterminer et soustraire les angles $\Delta\theta$ et $\Delta\phi$ du dessin. On dénote d la distance entre la blanche et la bille objet. On se souvient également que la distance entre la blanche et la fantôme de l'illustration (rose pointillé) est représentée par la variable r . Ainsi, on peut déterminer les angles recherchés :



On peut maintenant s'intéresser à la région qui délimite l'utilisation de cet angle coupé. En fait, il suffit de vérifier si la distance entre la blanche et la position fantôme corrigée (cercle bleu sur l'illustration) est plus petite que la distance entre la blanche et son ancienne position fantôme (non présente dans la figure). De plus, si cette première distance est plus petite que la distance entre la blanche et sa deuxième position fantôme (rose pointillé de l'image), alors le coup direct est tout simplement impossible. En termes mathématiques, la région est délimitée par :

$$\frac{2}{2}$$

Se pourrait-il que l'on doive corriger la position des deux positions fantômes ? Oui, mais il faudrait que l'angle soit très grand et que la blanche soit très près de la bille objet. Cette situation correspond à une bille objet à quelques millimètres de la poche, donc ce cas n'est pas intéressant.

4.3. Contrainte de bande

Si la position d'une des balles fantômes de la blanche est à moins d'un rayon de la bande, alors cet emplacement est inatteignable avec la blanche, il faut le corriger.

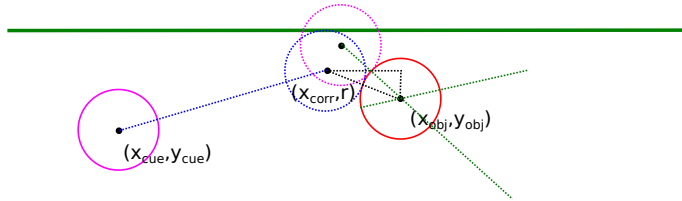


FIGURE 15. Balle fantôme contrainte par la bande

On cherche la position de la bille fantôme en bleu et on obtient :

$$\frac{2}{2}$$

On peut aisément déterminer , la distance entre cette position et la blanche, qui remplace la distance de la Section 3.1. Ensuite, on peut calculer , la distance entre cette position et celle de l'autre bille fantôme, afin de remplacer la quantité . Maintenant que l'on connaît , et , les dimensions du triangle

formé par la blanche et ses deux fantômes (dont l'une est corrigée), on peut utiliser la loi du cosinus pour déterminer l'angle de coup :

Si on tombe sur un cas où la bande restreint l'angle de tir et qu'en plus la blanche se trouve dans la région décrite à la fin de la section précédente, alors on peut tout simplement calculer le θ associé à chacun et choisir le plus petit des deux.

4.4. Rebond sur la bande

L'utilisation de la bande avec un coup de blanche est souvent à éviter. En effet, l'angle θ produit par un coup direct est généralement plus grand que celui formé par le rebond avec la bande. Néanmoins, il peut arriver que la collision avec le rebord de la table soit obligatoire, surtout si le coup direct est bloqué par des obstacles. Pour une position de bille fantôme P donnée et en se servant des calculs faits dans la Section 3.1, on obtient directement la position sur la bande :

On n'a qu'à faire ce calcul deux fois (une fois par bille fantôme), puis on peut obtenir les dimensions du triangle formé par la blanche et ses deux fantômes de bande, puis utiliser la loi du cosinus pour déterminer θ .

4.5. Combines

Si l'on veut calculer le θ résultant d'une combine, la technique est alors simple :

- À l'aide de la Section 3, on détermine l'angle θ .
- On calcule les positions fantômes adjacentes à la bille objet.
- On utilise l'analyse de la Section 4, mais en remplaçant la blanche par une seconde bille objet, afin de trouver un certain angle θ (qui correspond au θ de l'analyse).
- On calcule les positions fantômes adjacentes à la nouvelle bille objet.
- On utilise l'analyse de la Section 4 à nouveau, pour trouver l'angle θ .

Il est évident qu'en modélisant une combine, le marge d'erreur ϵ va être considérablement diminuée.

4.6. Remarque sur

Il est intéressant de noter que si la blanche est frappée dans la direction centrale de l'angle θ , alors, en général, après la collision avec la bille objet, cette dernière ne va pas rouler dans la direction du centre de l'angle θ . De plus, comme indiqué dans la section suivante, le centre de l'angle θ représente probablement la meilleure direction de coup. Ainsi, comme le milieu de la marge d'erreur de la blanche ne correspond pas nécessairement au milieu de la marge d'erreur de la bille objet, il n'est pas nécessairement avantageux de frapper la blanche dans le but de faire rouler l'autre bille vers le centre de l'entrée de la poche. Il s'agit, certes, d'une déduction simple de cette analyse, mais tout de même très importante à retenir lors d'un match de billard.

Résumé.

avec

,

et

Posons

avec

et

Coup direct	-	-	
Angle coupé	-	<div><div></div><div></div><div></div><div></div></div>	
Contrainte de bande	<div><div></div><div>2</div></div>	-	

On définit , le vecteur pointant vers la direction du centre de l'angle , à partir du centre de la blanche.

-

-

5. Probabilités

Une fois l'angle de marge d'erreur () pour la blanche calculé, on peut s'intéresser à déterminer comment maximiser nos chances de réussir un coup. On peut déduire, de l'analyse précédente, qu'il serait plus avantageux de frapper au centre de l'angle . En effet, cette stratégie laisse une marge d'erreur égale à gauche et à droite de la direction du coup. Cette erreur est difficilement modélisable, mais la loi probabiliste la plus logique à utiliser serait la loi gaussienne (normale). Le paramètre de moyenne est évidemment nul, car on veut qu'en moyenne, un joueur frappe dans la direction centrale de la marge d'erreur. Le paramètre de variance est par contre plus difficile à déterminer. Il indique à quel point, en moyenne, le joueur a tendance à s'éloigner de la moyenne. La seule façon d'approximer cette variable est avec un assez grand échantillon de coups.

Dans la dernière équation, — est la fonction erreur qui calcule la probabilité qu'une variable aléatoire de loi ² se retrouve entre et .

6. Conclusion

Bien que cet article fasse une analyse détaillée de la sensibilité des coups au billard, beaucoup de travail reste à faire. En effet, au début du document, beaucoup de facteurs ont été mis de côtés et beaucoup d'hypothèses ont été faites tout au long de l'article. Les rebonds sur les bandes, par exemple, ont été étudiés en tenant pour acquis que l'angle d'incidence était égal à l'angle

de réflexion, ce qui n'est pas le cas, en général. De plus, seuls les coups ayant des trajectoires linéaires ont été pris en compte, alors qu'en réalité les billes peuvent avoir des trajectoires curvilignes. Il est clair qu'en introduisant ce genre de coups, les marges d'erreur calculées vont être augmentées. En effet, si certains obstacles rendent impossible un coup en ligne droite, une trajectoire curviligne peut permettre de les contourner.

Les trajectoires curvilignes (lors du glissement de la bille) sont décrites par des équations [LG06, Mar95] qui dépendent du temps :

$$-\quad -$$

où \mathbf{p}_0 est la position initiale d'une bille sur la table, θ_0 est l'angle de direction du coup par rapport à l'axe des abscisses, v_0 est la vitesse initiale dans le sens du coup, μ est le coefficient de friction du tapis, g est l'accélération gravitationnelle et \mathbf{v}_0 est le vecteur de vitesse relative initiale au point de contact de la bille avec la table. Après un certain temps t_c , la bille commence à rouler et sa trajectoire devient linéaire.

Une étude intéressante serait d'analyser une fois de plus la sensibilité des coups, mais en mettant en évidence les trajectoires curvilignes. Ainsi, on pourrait déterminer les marges d'erreur non seulement avec la position des billes sur la table, mais aussi avec un vecteur de données initiales du coup (coordonnées de contact queue-blanche, direction du coup, angle vertical du coup et vitesse initiale).

Références

- [Alc04] D.G. Alciatore. *The Illustrated Principles of Pool and Billiards*. Sterling, 2004.
- [LDM11] Jean-François Landry, Jean-Pierre Dussault, and Philippe Mahey. Billiards : an optimization challenge. In *Proceedings of The Fourth International C* Conference on Computer Science and Software Engineering*, C3S2E '11, pages 129–132, New York, NY, USA, 2011. ACM.
- [LG06] Will Leckie and Michael Greenspan. An event-based pool physics simulator. In H. van den Herik, Shun-Chin Hsu, Tsan-sheng Hsu, and H. Donkers, editors, *Advances in Computer Games*, volume 4250 of *Lecture Notes in Computer Science*, pages 247–262. Springer Berlin / Heidelberg, 2006.
- [Mar95] W.C. Marlow. *The physics of pocket billiards*. MAST, 1995.

NICOLAS BUREAU, DÉPARTEMENT DE MATHÉMATIQUES, UNIVERSITÉ DE SHERBROOKE
 Courriel: Nicolas.Bureau@USherbrooke.ca

INTERPOLATION ET ESTIMATION DE DÉRIVÉES EN DIMENSION SELON UNE APPROCHE LAGRANGIENNE

ALEXANDRE DAY

RÉSUMÉ. L'interpolation lagrangienne, une approche de base en méthodes numériques, est bien adaptée à la représentation continue de données discrètes en dimension 1. Cet article a pour but de présenter l'extension de cette méthode en dimension arbitraire, ce qui permettra de calculer les dérivées partielles d'une fonction représentée par un nuage de points.

1. Introduction

Les méthodes d'interpolation et d'estimation de dérivées, à partir de données discrètes représentant une fonction, sont primordiales lors d'études de phénomènes physiques. En effet, ces méthodes trouvent de nombreuses applications dans la simulation de phénomènes astrophysiques ou cosmologiques [Abe11]. Les routines d'interpolation numérique étant peu onéreuses en terme de temps de calcul, elles sont souvent intégrées à des algorithmes plus complexes qui y font constamment appel. D'autres approches plus sophistiquées permettent d'approcher le problème comme la smooth particule hydrodynamics (SPH) [GM77], [Mon92]. Cependant, dans ce dernier cas, il est difficile d'évaluer l'erreur commise et la précision obtenue est limitée.

Tout d'abord, j'exposerai l'interpolation de Lagrange en dimension 1. Ensuite, je présenterai une notation qui sera utile pour démontrer une méthode d'interpolation en dimension n . Enfin, des résultats numériques seront présentés.

2. Interpolation de Lagrange [Bur]

Définition. Pour l'ensemble de couples (x_i, y_i) décrivant une fonction f , le problème consiste à trouver une fonction p telle que $p(x_i) = y_i$ pour $i = 0, \dots, n$. On dira alors que p interpole f aux noeuds x_i .

Plus particulièrement nous étudions l'interpolation polynomiale. En dimension 1, il existe un polynôme p de degré n unique qui interpole

J'aimerais tout d'abord remercier Jean-Marc Belley pour ses judicieux conseils durant le projet de stage. De même, je souhaite remercier Virginie Charette et le CRSNG pour leur support financier.

un ensemble de couples , lequel nous définissons comme le polynôme de Lagrange :

$$(1) \quad \text{---}$$

est bien un polynôme de degré . De plus, il interpole :

$$(2)$$

Où la définition de , le delta de Kronecker, est donnée plus bas (13). Démontrons maintenant l'unicité du polynôme.

THÉORÈME 1. *Soient couples . Alors, il existe un unique polynôme de degré tel que ,*

DÉMONSTRATION. Supposons qu'il existe deux polynômes de degré distincts et : qui interpolent couples représentant une fonction . Par définition, et doivent respecter la relation suivante :

$$(3)$$

Cependant, on constate que est au plus un polynôme de degré et possède donc au plus racines. Dans l'Éq. (3), il y a plutôt racines ce qui impose que la seule solution est .

La figure 1 illustre un exemple d'interpolation de Lagrange pour une fonction définie par morceaux et une fonction exponentielle décroissante oscillante. À gauche : on constate que malgré les discontinuités de la fonction étudiée, on réussit à en faire une bonne approximation dans la région la plus centrée de l'intervalle d'interpolation. L'usage d'un plus grand nombre de noeuds ainsi que l'application à une fonction continue augmente la qualité de l'interpolation comme il est illustré dans la partie de droite. L'interpolation est généralement instable aux extrémités de l'intervalle d'interpolation.

3. Notation multi-indice

Avant d'introduire l'extension de l'interpolation de Lagrange aux dimensions supérieures à 1, il convient d'introduire une notation concise pour des fins de clarté. Un multi-indice de longueur est un vecteur où les composantes sont des entiers non négatifs. Le multi-indice d'éléments tous nuls est noté . La norme d'un multi-indice est notée :

$$(4)$$

La dérivée d'un fonction sera notée :

$$(5) \quad \text{---} \quad \text{avec}$$

De plus, deux multi-indices de même longueur , sont égaux seulement si .

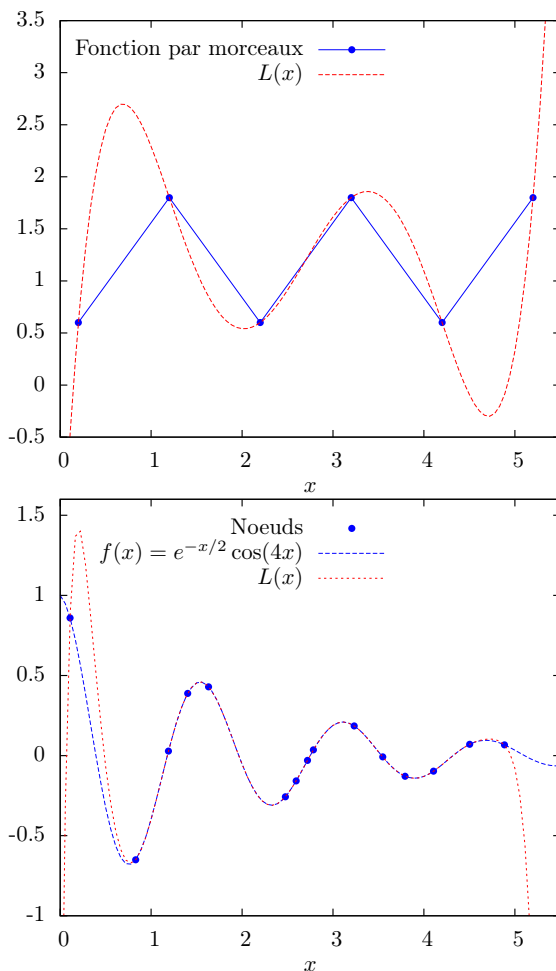


FIGURE 1. Interpolation par un polynôme de Lagrange (rouge). À gauche : fonction définie par morceaux pour une grille de 6 noeuds (en bleu) uniformément espacés. À droite : fonction indiquée sur la figure interpolée à partir d’une grille aléatoire de 15 points sur un intervalle

3.1. Domaine d’étude de l’interpolation

Soit un ensemble convexe Ω et soit $\mathcal{C}(\Omega)$, la classe des fonctions réelles de Ω qui admettent une extension continue et réelle sur un ensemble ouvert contenant Ω . On notera $\mathcal{C}^k(\Omega)$ ($k \geq 0$) la classe de fonctions k -fois continûment dérivables dans le voisinage de Ω , c’est-à-dire que pour tout multi-indice α tel que $|\alpha| \leq k$ tel que

(6)

la dérivée $\partial^\alpha f$ existe et est continue pour tout point $x \in \Omega$ dans un ensemble ouvert contenant Ω . Prenons $\mathcal{C}(\Omega)$ pour noter la classe de points (possiblement désordonnés) $\{x_i\}_{i=1}^n$, tel que $x_i \in \Omega$ et où $f(x_i)$ prend une valeur réelle en x_i . L’objet de cet article est de montrer que, pour un

arbitraire et un multi-indice respectant (6), il est possible de trouver des coefficients tels que la dérivée est approximée par le polynôme interpolant : défini comme :

(7)

Nous montrerons également que la connaissance de suffit à déterminer les polynômes interpolant les dérivées de :

(8) et

La façon de déterminer les coefficients est présentée à la section suivante.

4. Interpolation en dimension

4.1. Généralisation

Supposons que la fonction avec admet le développement (7), pour l'instant considéré comme exact. Rappelons également le développement de Taylor à plusieurs variables pour : Le premier terme, est le polynôme de Taylor de degré avec et et :

(9) _____

Le second terme est le reste de Cauchy qui ne fera pas l'objet du présent article. Il est cependant à la base de la façon d'évaluer l'erreur sur l'interpolation. Ainsi, en ne gardant que le polynôme de Taylor de degré , , nous substituons ce dernier dans (7), i.e. nous posons , :

(10) _____

(11) _____

Ainsi, si l'on souhaite estimer la dérivée partielle - où respecte (6) - en un point à partir du nuage de points , c'est-à-dire calculer , il suffit d'imposer que la deuxième somme de (11) soit nulle lorsque est différent de et égale à l'unité sinon. Cette condition s'exprime comme :

(12)

Où est le delta de Kronecker défini comme :

(13) si
autrement

Il s'agit dès lors de résoudre le système linéaire (12) afin de déterminer les coefficients a_{α} , qui pourront alors être utilisés dans (7). Si nous appliquons des deux côtés de l'égalité de (12), avec $\delta_{\alpha\beta}$ nous obtenons :

(14)

Mais en vertu (12), le deuxième terme peut se réécrire comme un delta de Kronecker :

(15)

D'où par unicité de (12) - dont la condition est énoncée à la section 4.2 - il s'ensuit que :

(16)

Cette relation est valide pour tout multi-indice α tel que $|\alpha| \leq n$ et donc, par induction, la relation (16) est satisfaite pour tout α tel que $|\alpha| \leq n$ respecte (6). Selon ces mêmes conditions, on obtient que :

(17)

4.2. Cardinalité de

On cherche ici à étudier un système linéaire dont la matrice des coefficients $M_{\alpha\beta}$ est carrée. Ainsi, selon la dimension et l'ordre de la dérivée que l'on souhaite estimer, seulement certaines cardinalités de α seront permises. Le nombre de multi-indices distincts satisfaisant l'inégalité (12) doit donc évaluer le nombre de coefficients $M_{\alpha\beta}$:

(18)

Où n est la dimension et $|\alpha|$ les indices définit dans (12). Bien-sûr, supposant cette dernière condition satisfaite, le système linéaire (12) aura une solution unique si le déterminant de la matrice des coefficients est non-nul.

5. Exemple numérique

Dans cette section nous prenons $n=2$. Tel qu'énoncé plus haut α sont des entiers non négatifs qui respectent (18). Nous choisissons un nuage de points distribués aléatoirement :

(19)

À partir de \mathcal{P} on construit l'ensemble à interpoler \mathcal{I} pour la fonction f . Nous calculons ensuite ∇f pour estimer la dérivée partielle ∇f au point \mathbf{x} . Les résultats sont présentés à la figure 2. Certaines des données de la figure 2 sont présentées dans le tableau 1. L'erreur (en bleu) converge exponentiellement (échelle log) pour le point estimé.

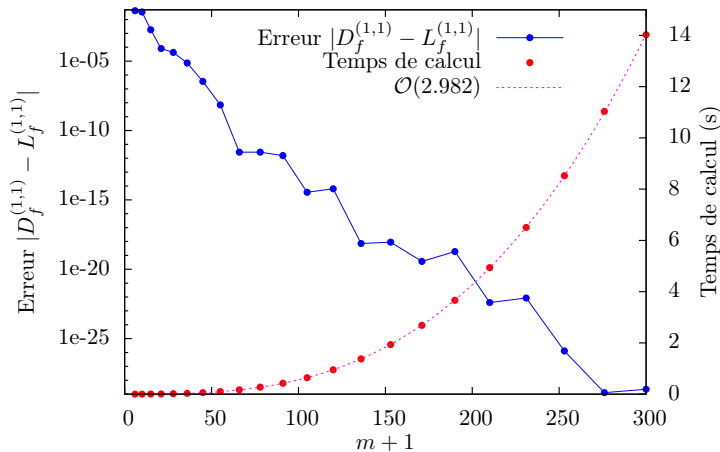


FIGURE 2. Erreur commise sur l'interpolation de (en bleu) sur une échelle log et temps de calcul (en rouge).

Cependant, le temps de calcul¹ (points rouges) augmente en puissance de 2.982 (en magenta) en fonction du nombre de points utilisés (). Cette complexité algorithmique est simplement la complexité typique reliée à la résolution d'un système linéaire général. Enfin, la grande précision obtenue dans les calculs a été rendue possible par l'utilisation de la librairie NTL, avec des types à précision quasi arbitraire. L'utilisation d'une grande précision devient particulièrement importante lorsque vient le temps de construire la matrice des coefficients.

TABLEAU 1
Exemple d'interpolation

Nombre de noeuds		Temps de calcul (s)	
2	6	0.043122	0.000636
3	10	0.035022	0.001519
5	21	0.79752e-4	0.00763
10	66	0.27465e-11	0.169808
16	153	0.89979e-18	1.93244
20	231	0.83731e-22	6.507
23	300	0.22257e-28	14.0281

Références

- [Abe11] T. Abel, *The first stars as seen by supercomputers*, Physics today, avril 2011
- [GM77] R.A. Gingold and J.J. Monaghan, Smoothed particle hydrodynamics : theory and application to non-spherical stars, Mon. Not. R. Astron. Soc., 181(1977), pp. 375-398
- [Mon92] J.J. Monaghan, Smoothed particle hydrodynamics, Annu. Rev. Astron. Astrophys., 30(1992), pp. 543-574.

1. Il convient de noter que les calculs ont été effectués sur un processeur standard de 2 GHz.

[Bur] R.L. Burden, *Numerical Analysis*, 9^e édition, 872 pages.

ALEXANDRE DAY, DÉPARTEMENT DE PHYSIQUE, UNIVERSITÉ DE SHERBROOKE
Courriel: `alexandre.day@usherbrooke.ca`

LES ARBRES DE DÉCISION HYBRIDES

ADAM SALVAIL-BÉRARD

RÉSUMÉ. Les bases de données, toujours grandissantes, renferment une variété d'informations qui n'attendent qu'à être extraites. Pour y parvenir, un éventail d'outils de forage de données a été inventé. Parmi ceux-ci, une méthode qui permet de créer de nouveaux algorithmes de forage est d'hybrider certains algorithmes existants. Cet article se penche sur les hybrides qui utilisent entre autre l'arbre de décision dans leur conception. L'arbre de décision est d'abord décrit en détail puis une exploration des méthodes d'hybridation est exposée. Enfin, un exemple d'arbre de décision hybride capable de prendre en charge les coûts de test et de mauvaise classification est présenté.

1. Introduction

Depuis quelques années, il est possible de constater une nette augmentation de l'information prête à la consommation et celle-ci ne cesse de croître. Grandement aidé par les outils informatiques maintenant à notre disposition, l'entreposage de données numériques est aujourd'hui plus facile que jamais. Par conséquent, il est routinier de garder une quantité importante d'informations, même si elle ne semble pas être une source, à première vue, de connaissances autres que celles représentées par les données elles-mêmes.

Pourtant, ces énormes bases de données sont des mines d'informations d'une grande valeur pour ceux qui sont en mesure d'en extraire les secrets. La discipline qui se spécialise dans cette extraction s'appelle le forage de données (*data mining*)^{1 2}.

L'une de ces techniques d'extraction, l'arbre de décision, retient l'attention tant par sa simplicité que par la facilité d'interprétation des modèles qu'il crée. Cet article, après une présentation des concepts utiles à la lecture de ce document,

J'aimerais remercier chaleureusement le CRSNG pour son appui financier et Jessica Lévesque pour ses conseils et son soutien au niveau de la rédaction.

1. Il est courant de rencontrer les termes « exploration des données », « fouille de données » ou même « extraction de connaissances ». En anglais, il est fréquent de rencontrer également « *Knowledge Discovery in Data* ».

2. Le domaine du forage de données, comme bon nombre de domaines des technologies, utilise le plus souvent l'anglais pour transmettre ses informations, notamment dans le choix de nom des nouveaux concepts. C'est pourquoi ce rapport indique les termes anglophones en *italique* de la plupart des définitions. Prendre note également que certaines définitions francophones sont le fruit d'une traduction libre.

présente l'arbre de décision dans le but de l'améliorer en créant des techniques d'apprentissage hybrides. Finalement, un exemple d'algorithme hybride capable de prendre en charge les coûts de test est présenté.

2. Le forage de données

Cette section fait le point sur les connaissances requises afin de favoriser la compréhension des sections suivantes. Il est question de parcourir quelques concepts de base du forage de données et de se familiariser avec les notions et le vocabulaire propres à ce domaine.

2.1. Les méthodes d'apprentissage

Une base de données est une collection de mesures qui représentent un système. Ces mesures sont encodées sous forme de **données** qui sont la représentation de ces mesures sur un support informatique. Ces données sont entreposées dans une base de données sous forme d'**instances**³ qui sont des vecteurs de valeurs fournies par des mesures. Chaque instance représente un élément de l'étude. Avec ces définitions, une **base de données** devient donc une collection d'instances.

Par exemple, pour une base de données contenant des informations sur un ensemble de figures géométriques, chaque figure est une instance de la base de données. De plus, chacune de ces figures (ou instances) possède des caractéristiques (hauteur, largeur...) appelées **attributs** ou **variables**. Ces valeurs sont les mesures qui représentent chaque instance. Dans un autre domaine, un foreur de données peut s'intéresser à l'influence de la météo sur l'achalandage d'un parc d'attractions : une instance pourrait être une journée d'activités et représentée par divers attributs concernant la météo, l'achalandage, le chiffre d'affaires... Ou encore, il est possible d'étudier la généalogie des vaches laitières : les vaches sont les instances et les attributs sont les informations qu'il est possible de recueillir sur chacune des vaches. Dans le domaine de la finance, il est possible de lister de nouvelles entreprises (les instances) selon plusieurs caractéristiques (les attributs) pour les classer ou prédire leurs chances de réussite. Pour les banques, chaque client de leurs bases de données est un individu sur lequel ils possèdent une panoplie d'informations les aidant à faire des études pour, par exemple, attribuer les cotes de crédit.

Formellement, chaque instance est donc un vecteur de dimension égale au nombre d'attributs d'une base de données. Un attribut est dit **catégorique** ou **numérique**. De façon simple, si la moyenne des valeurs a un sens, il s'agit d'un attribut numérique. Sinon, l'attribut est catégorique. Par exemple, la couleur d'un objet est un attribut catégorique alors que sa hauteur en centimètres est un attribut numérique.

Les techniques de forage de données sont séparées en deux grands groupes. Le premier, **l'apprentissage non supervisé**, regroupe les procédures qui tentent de décrire l'organisation des données. Un exemple d'algorithme d'apprentissage non supervisé est **l'analyse de groupement** dont le but est de grouper les instances semblables selon leurs similarités.

3. Une instance est aussi souvent appelée un enregistrement, une observation ou un individu.

Le deuxième grand groupe de méthode d'apprentissage concerne l'**apprentissage supervisé**. Dans cette catégorie, le but est souvent de décrire l'information (graphiquement, à l'aide de règles...) en créant un **modèle** représentatif des données étudiées. Un tel modèle sert à établir des liens entre un attribut d'intérêt, la variable **dépendante**, et les autres attributs, les variables **indépendantes**⁴.

Pour effectuer un apprentissage supervisé, le foreur de données dispose d'un certain nombre d'instances dont la valeur de la variable dépendante est déjà connue. C'est à partir de ces exemples que le modèle est construit et celui-ci sert ensuite à établir des prédictions. Partant du principe qu'il existe un modèle décrivant parfaitement la réalité, le but devient donc d'estimer le plus fidèlement possible ce modèle optimal.

De ces modèles découle le côté prédiction de l'apprentissage supervisé. Il fournit des méthodes pour estimer la valeur de la variable dépendante en se basant sur la valeur d'autres attributs et sur le modèle représentant les données. Ce pronostic est souvent le but ultime de l'utilisation des techniques d'apprentissage supervisé. C'est également ce qui est étudié dans cet article.

La prédiction de la variable dépendante est définie comme étant une **classification** ou une **estimation** si la nature de la variable dépendante est respectivement catégorique ou numérique. Dans le cas d'une classification, les valeurs possibles de la variable dépendante sont appelées les **classes**.

Un exemple de classification : il faut classer des figures géométriques selon leur forme (carré, rectangle, triangle...). Dans ce cas, chaque instance possède des attributs : nombre de côtés, longueur des côtés, aire de la surface, forme... Parmi ces attributs, la variable dépendante est la forme, car c'est sur la valeur de cet attribut que la technique d'apprentissage doit faire sa prédiction. Donc, « carré », « rectangle » et « cercle » sont tous des exemples de classes.

Pour le reste de l'article, l'accent est mis sur les méthodes de classification. Si une variable dépendante est numérique, elle serait au préalable **discrétisée**, c'est-à-dire que les valeurs de la variable dépendante seraient partitionnées, les parties devenant les nouvelles valeurs possibles de la variable transformée⁵. Par exemple, il est possible de discrétiser la variable « âge » en tranches d'âge ou le salaire d'un individu en classes salariales.

2.2. L'évaluation de l'apprentissage

Un modèle n'est pratiquement jamais parfait : il n'est pas en mesure de décrire entièrement le système sous-jacent. Les raisons sont multiples, mais le résultat est le même : une perte de fiabilité du modèle. Dans tous les cas, il faut pouvoir faire l'évaluation de la fiabilité du modèle créé. Pour ce faire, il existe différentes techniques.

La **performance** de l'apprentissage est donc une mesure qui représente la capacité du modèle créé à bien représenter un système. Pour estimer la performance, il faut procéder à certains tests et recueillir des mesures la décrivant. Une méthode populaire est de séparer les instances en deux groupes : les **données**

4. Par abus de langage, les variables indépendantes sont souvent appelées attributs sans autres précisions. Dans ce contexte, un modèle est formé d'attributs et d'une variable dépendante.

5. Pour plusieurs algorithmes, il est également nécessaire de discrétiser les attributs afin que l'algorithme puisse être appliqué.

d'apprentissage (ou **d'entraînement**), et les **données tests**. Les données d'apprentissage servent à la construction du modèle alors que les données test sont utilisées pour vérifier la fiabilité du modèle construit.

Les données d'apprentissage regroupent les instances desquelles l'algorithme choisi doit apprendre et construire son modèle. Une fois l'apprentissage terminé, une classification des données est accomplie. Cette classification s'opère en notant la valeur de la variable dépendante prédite par le modèle selon la valeur des variables indépendantes de l'instance à classer. Il est ensuite possible de déterminer un certain **taux d'erreur**, donné par le ratio du nombre d'instances mal classées sur le nombre d'instances étudiées.

Aussi, il peut être intéressant de calculer le taux d'erreur sur chacune des classes individuellement : peut-être que le modèle ne classe incorrectement que l'une d'elles. Cette information peut être particulièrement importante si le coût de mauvaise classification varie en fonction des classes.

Imputer des coûts aux bonnes et mauvaises classifications permet d'avoir un meilleur contrôle sur l'importance de celles-ci. Il est possible de définir une **fonction de coût** qui renseigne sur le coût d'une classification et une autre fonction qui indique le nombre d'instances classées d'une certaine façon. Par exemple, dans la table 1, le coût de la mauvaise classification d'une instance qui aurait dû être classée A mais qui a été classée B est donné par c_{AB} . Pour déterminer la performance d'un classificateur, il suffit de calculer le coût de classification d'un groupe d'instances :

où n_{ij} représente le nombre d'instances ayant comme classe i et classées dans la classe j .

		Valeurs prédites		
		A	B	C
Valeurs réelles	A			
	B			
	C			

TABLE 1. Exemple d'une matrice de coûts de dimension 3.

3. L'arbre de décision

Cette section présente en plus de détails le fonctionnement de l'algorithme qui sert à la création des arbres de décision. Avant d'entrer dans les détails, il faut d'abord se familiariser avec le vocabulaire lié aux arbres de décision. Ensuite, une présentation de l'algorithme général est donnée pour terminer avec les avantages et les inconvénients de cette méthode.

3.1. Définitions

Parmi les algorithmes de classification, l'un des plus simples d'utilisation et d'interprétation, tout en gardant des performances très respectables, est l'**arbre de décision** [WF05]. Existant sous plusieurs formes, l'arbre de décision est reconnu par le résultat de l'algorithme qui produit un modèle constitué d'un ensemble de règles de classification qu'il est possible de représenter sous forme d'**arbre** comme à la figure 1.

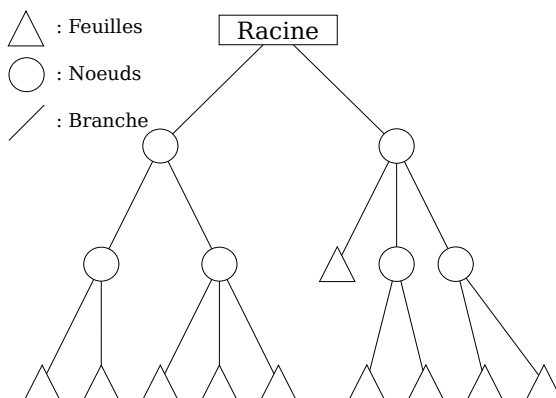


FIGURE 1. Un exemple d'arbre de décision qui illustre chaque composante de l'arbre.

Sur cette figure, il y a d'abord la **racine** de l'arbre qui se trouve tout en haut. Les instances à classer sont insérées dans l'arbre à partir de la racine. Elle contient l'ensemble des données d'entraînement et doit donner une première séparation qui amènera à la classification.

Dans l'exemple de la figure 2, pour chaque instance à classer, il faut regarder d'abord le nombre de côtés. Selon la valeur du nombre de côtés (plus grand que ou plus petit ou égal à), il faut suivre la **branche** correspondante. Par exemple, pour classer une figure à quatre côtés, il faut suivre la branche « » pour arriver au **noeud** « largeur ».

Un noeud contient l'ensemble des instances qui satisfont aux critères de séparation précédents. Ici, le noeud « hauteur » le plus élevé de l'arbre contient donc l'ensemble des instances qui ont une valeur « » à l'attribut « nbCotes » et une valeur « » à l'attribut « largeur ». Les noeuds, comme la racine, donnent d'autres critères de segmentation. En fait, la racine n'est qu'un noeud spécial : c'est le premier rencontré et il est le point d'entrée de toutes les instances.

Supposons maintenant que la figure, en plus de posséder quatre côtés, ait également une largeur de 4 et une aire de 8. Au bout du **chemin** construit à partir des branches qui satisfont aux caractéristiques de cette figure se trouve une **feuille**, nommée ainsi parce qu'elle-même n'a pas de branches. Une feuille est spéciale : chaque instance qui s'y trouve, ayant suivi le chemin de la racine jusqu'à la feuille, est classée de la même façon par l'arbre de décision.

La classification d'une nouvelle instance se fait à l'aide de l'arbre : il faut trouver la feuille de l'arbre qui contient l'instance à classer selon le critère de

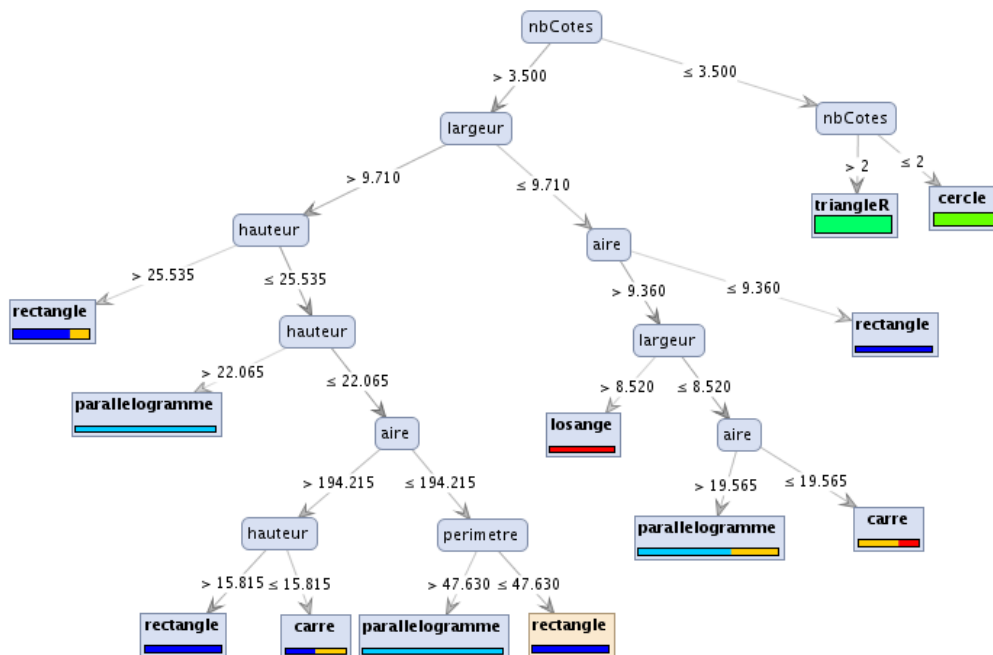


FIGURE 2. Exemple d'un arbre de décision .

segmentation et attribuer à l'instance la classe la plus fréquente des autres instances de cette feuille. Par conséquent, selon l'exemple, la figure décrite plus haut se retrouve dans la feuille étiquetée « rectangle » : elle est donc classée comme étant un rectangle.

La figure 3 donne un aperçu du vocabulaire lié aux arbres. Par exemple, le noeud précède le noeud et , est donc le **parent** de et est le **fil** de . De plus, il est possible de considérer comme étant la racine d'un **sous-arbre** de l'arbre principal. et sont également les **enfants** de . Aussi, est le frère de .

3.2. Algorithme de création du modèle

Les arbres de décision sont apparus simultanément dans deux disciplines, soit en statistiques et en apprentissage automatique [HK06] [WF05]. J. Ross Quinlan fut celui qui travailla sur les arbres de décision, au cours des années 70 et au début des années 80, du point de vue de l'apprentissage automatique. Il donna naissance à l'algorithme *ID3* [Qui86] et à ses successeurs le *C4.5* [Qui93] et le *C5.0* qui devinrent une référence sur laquelle les autres se basent aux fins de comparaison [HK06]. *C5.0* étant sous licence propriétaire et son code source n'étant pas disponible, les chercheurs continuent à se baser sur le *C4.5*⁶ [WF05].

Dans les mêmes années, L. Breiman, J. Friedman, R. Olshen et C. Stone ont développé la version statistique de l'arbre de décision en inventant l'algorithme *CART* [BFOS84]⁷.

6. Plus particulièrement la révision 8 du *C4.5* ou *C4.5R8* publiée par Quinlan.

7. D'autres algorithmes d'arbre de décision d'intérêt : QUEST, CHAID, SLIQ et SPRINT.

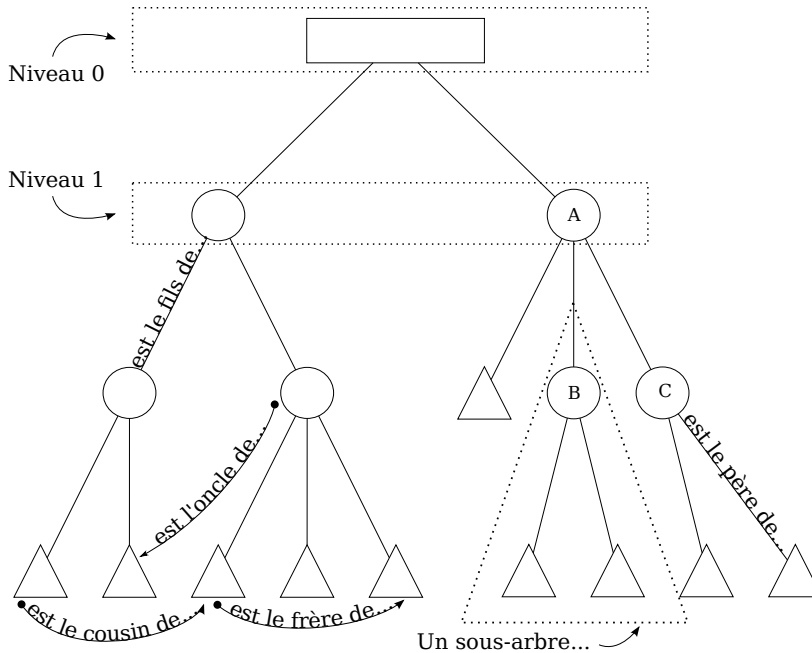


FIGURE 3. Illustration du vocabulaire relatif aux arbres.

Le fonctionnement de l'arbre de décision est simple, c'est ce qui le rend si facile d'interprétation et si populaire. Son apprentissage se fait par étape. À chaque étape, un noeud est divisé en plusieurs enfants selon une condition sur les valeurs d'un attribut. Pour ce faire, il faut avoir un moyen de trouver avec quel attribut créer la segmentation, c'est-à-dire la division du noeud. Pour ce faire, il existe plusieurs **critères de segmentation** qui servent à diviser les données qui se trouvent dans un noeud. Ce critère choisit de façon judicieuse un attribut dans le but d'avoir le meilleur modèle de classification possible. Dans l'exemple illustré à la figure 2, l'algorithme a décidé de séparer la racine selon l'attribut « NbCotes » en segmentant les figures ayant moins de côtés et ceux en ayant au moins .

L'un de ces critères de segmentation est le gain en information basé sur l'entropie de Shannon^{8 9} décrit par Claude Shannon vers la fin des années 1940 et représente en quelque sorte l'information véhiculée par un certain support (ici un attribut) [HK06].

3.3. Avantages

Un principal avantage de l'arbre de décision est sa capacité à représenter un système sous forme d'arbre ou d'une suite de règles de classification des données. Cette représentation du modèle illustre, en plus d'une classification, la logique derrière le modèle qui est en elle-même une information qui peut s'avérer intéressante [WF05].

8. Sans autre précisions sur l'algorithme, cet article fait référence à l'algorithme C4.5 comme algorithme d'arbre de décision par défaut. Cet algorithme utilise le critère de l'entropie de Shannon pour son critère de segmentation.

9. Quelques autres critères intéressants : l'indice de Gini, le et leurs dérivés.

De plus, il devient très facile de classer les nouvelles instances une fois l'arbre construit. En effet, comme l'apprentissage est effectué lors de la construction de l'arbre, une fois celui-ci terminé, chaque instance à classer n'a qu'à suivre un chemin de l'arbre, ce qui est très efficace [WF05].

Aussi, puisque l'algorithme reste assez simple, il est souvent privilégié par rapport aux autres méthodes plus lourdes à mettre en place, d'autant plus qu'un arbre bien construit donne des performances similaires à d'autres techniques plus complexes sur de petits ensembles de données [RN94] [WF05].

3.4. Inconvénients

La segmentation en de plus en plus petits ensembles de données pendant l'apprentissage amène un problème : la fragmentation des données. Une fois l'arbre rendu à quelques niveaux, il doit travailler sur un noeud contenant un nombre restreint d'instances, laissant place à un biais plus important dans ses prédictions [RN94].

Un autre désavantage est la tendance des algorithmes d'arbres de décision à favoriser la séparation des données en d'importants groupes en ignorant, ou reléguant à plus tard, la classification d'un groupe marginal de données. Or, ces groupes marginaux peuvent être tout aussi intéressants, sinon plus, que la structure générale des données [CF00].

Dans le même esprit, certaines fonctions de gain en information tendent à être biaisées envers les sous-ensembles de grosseurs semblables et multiples, ce qui ne reflète pas toujours la réalité [RN94].

Concernant la séparation d'arbres en sous-arbres, il peut se produire un problème de réplification des sous-arbres. Si un attribut sépare les données en deux groupes, il est possible, et même fréquent que chacun des sous-groupes puisse être redivisé selon une même série d'attributs, conduisant à deux sous-arbres avec la même structure et introduisant de la redondance dans le modèle [RN94].

Il est aussi possible d'imaginer comment un arbre pourrait être très différent si un attribut opérant une séparation au début de l'arbre était remplacé par un autre. Modifier quelques instances d'un échantillon d'entraînement peut avoir des effets importants sur la construction du modèle, par exemple en changeant le premier attribut choisit par l'arbre, ce qui le rend instable aux changements [WF05]. Le problème pourrait provenir seulement du choix des instances constituant les données d'entraînement. L'arbre est donc sensible à la façon d'échantillonner. Une façon de repérer ce problème est d'utiliser des techniques comme la validation croisée pour réduire l'effet de l'échantillonnage sur la mesure des performances de l'arbre ou une technique d'ensemble (voir section 4.1) pour diminuer l'importance de l'échantillonnage dans la classification des données.

L'un des pires défauts de l'arbre de décision est son incapacité de prendre plus d'un attribut en compte à la fois [WF05]. Bien qu'il existe des algorithmes d'arbre de décision permettant de découvrir des relations linéaires entre les variables, les algorithmes d'arbre de décision communément utilisés ne sont pas en mesure de répondre à ce problème. Par exemple, il est montré à la figure 4 un exemple où un arbre de décision complique beaucoup la classification comparé à la solution « simple » qui consisterait à tracer une droite séparant les données.

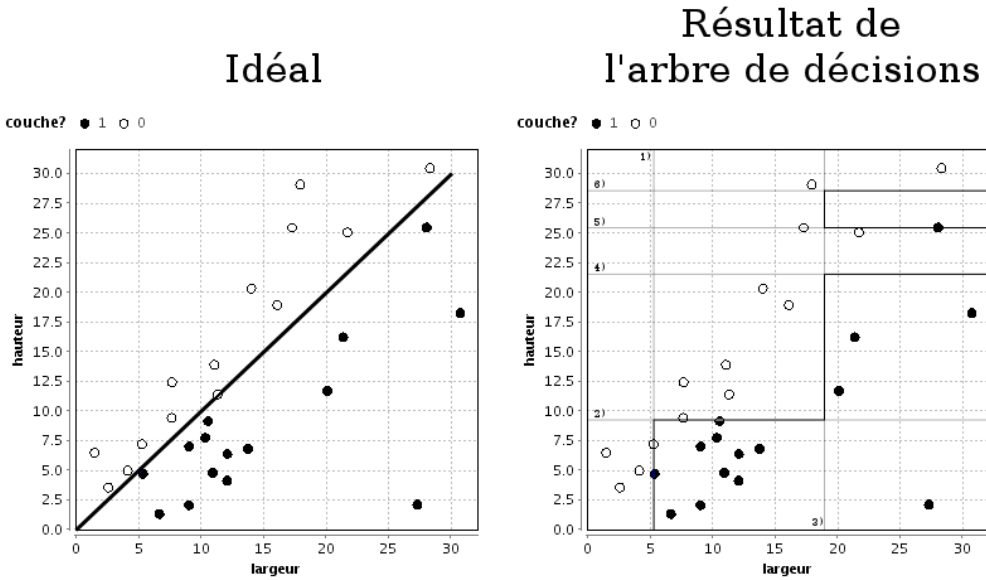


FIGURE 4. Cas où une séparation par une fonction linéaire serait plus appropriée, difficile à effectuer avec un arbre de décision.

Un algorithme, tel le C4.5, fait une segmentation ne s'appuyant que sur un attribut à la fois. Ceci donne des frontières de séparation constantes, c'est-à-dire une fonction constituée de lignes horizontales ou verticales seulement. Ceci n'est pas toujours idéal comme illustré à la figure 4 où une droite aurait suffi.

Ces inconvénients sont les principales motivations du reste de ce rapport. Les techniques hybrides ont été développées dans le but d'atténuer les conséquences négatives de l'arbre de décision tout en renforçant ses points positifs [AH98]. Plusieurs de ces défauts sont donc adressés dans les sections suivantes sur la combinaison de méthodes d'apprentissage.

4. Combiner l'arbre de décision à d'autres techniques

Alors que les arbres de décisions simples sont la plupart du temps basés sur les mêmes principes de segmentation des données, plusieurs lacunes au fil du temps sont ressorties. C'est dans l'objectif de réduire l'impact de ces défauts que les chercheurs ont voulu modifier de façon fondamentale l'algorithme, tout en essayant de garder la particularité qui le rend si populaire : sa facilité d'interprétation.

Pour y parvenir, quelques techniques ont été mises de l'avant. Parmi celles-ci, il est possible de retrouver les techniques d'apprentissage d'ensembles, et les algorithmes hybrides¹⁰.

10. Plusieurs classifications des algorithmes hybrides sont proposées dans la littérature. Notamment, Goonatilake et Khebbal [GK95] proposent une classification issue du domaine de l'intelligence artificielle que Anand et Hughes [AH98] réfutent pour son application au forage de donnée.

La suite de l'article fait l'objet, entre autres, de techniques de combinaison entre un arbre de décision et une tout autre méthode de forage de données comme la machine à vecteurs de support, le réseau de neurones, l'algorithme bayésien naïf ou l'algorithme génétique. Ces autres méthodes sortant du cadre de ce document, le lecteur désireux d'en savoir plus est invité à consulter les références de l'article, tout particulièrement [HK06] et [WF05].

4.1. Les techniques d'ensembles

Une première tactique utilisée afin de combiner les résultats de plusieurs itérations d'un même algorithme ou même de plusieurs algorithmes différents consiste à faire l'agrégation de différents modèles, c'est-à-dire de tenter de les combiner en un plus robuste et performant. Cette technique est appelée l'**apprentissage par ensembles** (*ensemble learning*) [WF05].

Parmi les techniques d'apprentissage par ensembles les plus populaires se retrouvent l'**agrégation par rééchantillonnage** (*bagging* qui vient de *bootstrap aggregating*) et le **renforcement** (*boosting*). Chacune de ces techniques fonctionne selon le principe de combinaison de plusieurs cycles d'apprentissage afin d'arriver à un modèle plus robuste. Voici une brève explication de leur fonctionnement.

L'**agrégation par rééchantillonnage** utilise plusieurs itérations d'un même algorithme comme l'arbre de décision. Dans le but de créer des modèles différents, les arbres sont construits à partir des données d'entraînement initiales échantillonnées avec remise. Il résulte donc d'un certain nombre d'ensembles de données d'entraînement, tous ayant le même nombre d'instances que l'ensemble de données d'entraînement initial. L'arbre de décision étant un algorithme plutôt instable, les perturbations entre chacun des échantillons sont suffisantes pour créer des modèles différents d'une itération à l'autre [WF05]. C'est donc une façon de se servir d'une faiblesse de l'arbre de décision à son avantage. Lors de la classification des nouvelles instances, chaque instance est classée dans tous les modèles et la classe la plus fréquente (pondérée selon la performance des arbres individuels ou non) est choisie comme étant la classification finale [WF05].

De l'agrégation par rééchantillonnage découle une technique très efficace et qui gagne en popularité appelée **forêt d'arbres décisionnels** (*random forests*). Cette technique, inventée par Breiman [Bre96] (l'inventeur du modèle CART, l'un des premiers arbres de décision), commence à faire sa place dans l'arsenal des foreurs de données.

L'algorithme de la forêt d'arbres décisionnels construit un nombre donné d'arbres de décision à l'aide d'un sous-ensemble restreint d'attributs pour chaque arbre dans le but d'insérer un effet aléatoire. Les instances sont toujours choisies au hasard et avec remise.

Ensuite, comme pour l'agrégation par rééchantillonnage, une instance est classée par chacun des arbres et la classe résultante la plus fréquente est utilisée pour la classification.

La seconde technique d'apprentissage par ensemble est le **renforcement**. Cette technique tente d'améliorer chacune des itérations de l'algorithme en mettant un poids plus important sur les instances mal classées pour les itérations suivantes. De cette façon, chaque fois qu'un arbre est reconstruit, la priorité de

classification des données mal classées précédemment est plus élevée, ce qui force l'algorithme à tenter de classer ces instances correctement le plus tôt possible. Contrairement à l'agrégation par rééchantillonnage où le nombre d'itérations est arbitraire, ici certaines conditions d'arrêts sont définies selon le taux d'erreur du modèle produit [WF05]. Le lecteur intéressé peut en apprendre davantage en s'informant sur l'algorithme AdaBoost [FS95], l'implantation la plus populaire du renforcement.

4.2. Tri des attributs

Le tri des attributs peut améliorer significativement la performance d'une méthode d'apprentissage. Il a été adopté par plusieurs auteurs pour l'hybridation d'un arbre de décision, dans le but d'améliorer les performances de classification.

Par exemple, dans [WF05] il est proposé d'utiliser un arbre de décision afin de faire un premier tri des attributs et de ne retenir que ceux utilisés par le modèle résultant. Bien sûr, il est inutile de faire cela si la méthode d'apprentissage est elle-même un arbre de décision.

C'est le modèle utilisé par Kissi et Ramdani [KR10] qui propose d'utiliser un arbre de décision pour faire une première sélection des attributs. Les attributs restants sont ensuite fournis à un réseau de neurones qui s'occupe de faire la classification de façon usuelle.

Alors qu'il est possible d'utiliser un arbre de décision avant un réseau de neurones, rien n'a empêché Fong et coll. [FSBA09] de faire l'inverse pour leur problème de prédiction de l'admission dans les universités. De plus, comme c'est l'arbre de décision qui s'occupe de fabriquer le modèle de classification, l'algorithme retient l'avantage principal de l'arbre de décisions : sa facilité d'interprétation. Leur algorithme, appelé RSAU, utilise encore une fois le principe de sélection des attributs pour améliorer la performance de leur modèle. Cette fois, c'est le réseau de neurones qui a pour objectif de faire un premier tri des attributs pour permettre au C4.5 de faire un modèle de prédiction plus performant.

Dans un article de Nair et coll. [NMS10], les auteurs utilisent un arbre de décision afin de faire un premier tri des attributs avant de laisser le soin aux esquisses d'ensemble (*rough set*) de faire la classification.

Le tri des attributs avant l'apprentissage est une étape importante. Par exemple, une bonne raison de procéder à un tel tri est qu'il a été démontré que l'ajout d'un attribut binaire provenant de tirs d'une pièce non biaisée fait décroître les performances du modèle C4.5. Les performances résultantes affichent un taux d'erreur de 5 à 10 % plus élevé [WF05]. Le tri des attributs par un second algorithme d'apprentissage différent est donc une bonne façon d'éliminer le bruit causé par des attributs superflus.

Bala et coll. [BHV 95] introduisent une nouvelle technique pour faire un tri préventif des attributs fournis à l'algorithme de construction de l'arbre de décision à l'aide d'un algorithme génétique. L'idée est reprise par Stein et coll. [SCWH05] et Kornienko et coll. [KB03] mais avec quelques nuances.

L'analyse en composantes principales¹¹ peut également être utilisée conjointement avec l'arbre de décision afin de s'assurer que les attributs étudiés couvrent

11. Pour plus d'informations au sujet des analyses en composantes principales et leur importance dans le domaine du forage de données, une bonne introduction est donnée dans [WF05].

le plus de variance (donc d'information) possible afin de permettre à l'algorithme de l'arbre de décision de faire des segmentations qui ne sont pas nécessairement parallèles aux axes originaux. En reproduisant une analyse en composante principale à chaque noeud, il est possible d'arriver à un arbre de décision multivarié, c'est-à-dire qui prend compte de plus d'une variable à la fois [WF05].

4.3. Partition des instances

Un autre objectif recherché par les auteurs d'arbres de décision hybrides est d'effectuer une partition des instances, souvent pour faire en sorte qu'une partie des instances ne soit pas classée par le même algorithme. La séparation en multiples parties permet, par exemple, d'utiliser un algorithme rapide, mais moins performant en combinaison avec un algorithme performant, mais moins rapide.

C'est l'approche utilisée par Kumar et coll. [KG10] qui tentent d'améliorer le temps de classification d'une machine à support de vecteur en classant les instances facilement classables par l'arbre de décision et reléguant les cas plus délicats à la machine à vecteurs de support.

Comme il a déjà été mentionné, un arbre de décision est une méthode efficace si utilisée avec des variables catégoriques. Par contre, l'utiliser pour la classification d'une variable numérique implique une discrétisation de la variable au préalable pour l'arbre de décision de Quinlan [Qui93]. Or, c'est l'inverse pour un réseau de neurones : une variable numérique n'a aucun mal à être utilisée par le modèle, alors que pour les variables catégoriques l'algorithme doit utiliser des stratagèmes pas toujours efficaces pour la classification des données. Les mêmes restrictions s'appliquent sur les attributs utilisés pour l'apprentissage.

C'est en conséquence de ceci que Zhou et Chen [ZC02] ont inventé HDT, un algorithme hybride qui utilise un arbre de décision pour gérer l'information des variables indépendantes catégoriques et un réseau de neurones pour s'occuper des variables indépendantes numériques.

Une approche similaire a été adoptée par Arentze et Timmermans [AT07] qui utilisent un modèle logit ¹² afin de faire la séparation d'un attribut numérique.

Tsai et Wang [TW09], quant à eux, filtrent les instances avec un réseau de neurones : seules les instances bien classées sont utilisées comme données d'apprentissage pour bâtir l'arbre de décision.

4.4. Modification du comportement des feuilles

Les hybrides de cette section visent à modifier la façon dont une instance est classée. Par exemple, l'algorithme C4.5 effectue le classement d'une instance en prenant, dans une feuille, la classe la plus fréquente. C'est ce principe qui est remplacé dans cette section.

Pour des raisons de performance, Kohavi [Koh96] et Wang et coll. [WZX10] ont décidé d'utiliser l'algorithme bayésien naïf pour continuer le travail de l'arbre de décision une fois l'arbre élagué. En effet, pendant le processus d'élagage, l'algorithme bayésien naïf est utilisé afin de tester si l'application de cet algorithme

Pour une analyse plus approfondie, il peut être intéressant de se référer à un ouvrage statistique sur le sujet.

12. Plus d'informations sur le modèle logit peuvent être obtenues en cherchant une description de la régression logistique, décrite entre autres dans [WF05].

dans un noeud précédant les feuilles améliore les performances du classificateur. Une fois l'arbre élagué, il y a donc certaines feuilles qui ont effectivement une autre technique totalement implantée à partir des instances disponibles à ce point.

Dans leur article, Seewald et coll. [SPW01] ont pris l'idée de Kohavi [Koh96] et l'ont étendue. Au lieu de seulement remplacer le classificateur standard par un classificateur bayésien, ils ont laissé le choix à l'utilisateur entre trois classificateurs : la méthode bayésienne naïve, IB1 (un algorithme du plus proche voisin) et la régression linéaire.

Bien que ce ne soit pas l'algorithme lui-même qui fait le choix de la méthode, il est concevable que chacune d'elles puisse être testée et que seule la plus performante dans chaque noeud soit retenue.

La classification par régression linéaire est décrite par Frank et coll. [FWI 98] dans leur article sur les arbres modèles (*model trees*).

Ce principe rappelle la régression logistique (une fonction qui minimise l'erreur entre les données et une courbe logistique) qui pourrait être mixée avec un arbre de décision. En effet, c'est ce que Landwehr et coll. [LHF03] ont utilisé dans leur article.

Cet algorithme a été produit dans le but de diminuer le biais de la classification en rendant possible la classification locale des données. Ici, l'algorithme C4.5 ne sert qu'à subdiviser l'espace des solutions en régions sur lesquelles est appliquée une méthode d'apprentissage locale.

Le raisonnement par cas, une autre méthode, peut être coûteux en ressources. Dans le but de vaincre ce problème, Bouyer et coll. [BAM07] utilisent un arbre de décision pour localiser l'apprentissage, ce qui le rend plus performant et plus rapide à exécuter.

Suivant une autre idée, il est possible de remarquer que lors de la construction d'un arbre de décision, l'algorithme a tendance à privilégier les séparations en groupes importants, c'est-à-dire qui contiennent beaucoup d'instances. Les plus petits groupes sont souvent mal classés puisque du point de vue de l'algorithme ils sont plus propices à refléter une classification erronée, surentraînée ou tout simplement causée par du bruit dans les données d'apprentissage.

Or, il se trouve que souvent l'information intéressante et inconnue se trouve être une conséquence de l'existence de petits groupes d'instances qui sont souvent mis de côté dans le processus de décision. Le problème est de savoir comment les reconnaître de façon efficace.

Carvalho et Freitas [CF00] [CF04] se sont penchés sur la question et ont décidé de recourir à un algorithme génétique afin de régler le problème. L'idée est tout simplement de recourir au principe d'évolution de l'algorithme génétique sur l'ensemble des instances qui tombent dans une feuille avec un faible nombre d'instances afin de leur attribuer des règles de classification plutôt que de les élaguer hâtivement. L'esprit de cette méthode a été repris dans les travaux de Noubi [Nou].

Sur un autre plan, l'utilisation de la conversion du modèle de l'arbre de décision en règles de classification est utilisée par Pistori et Neto [PN03] qui ont développé un hybride entre un arbre de décision et la théorie des automates. Une des caractéristiques de cet algorithme nommé *Adaptree* est qu'il tente de changer

l'ordre d'apparence des attributs dans les règles en utilisant quelques stratégies syntaxiques.

4.5. Altération de la gestion des coûts

Comme discuté dans la section 2.2 sur l'évaluation des performances des méthodes d'apprentissage, une pratique efficace est d'affecter des coûts aux différentes classifications correctes ou erronées. Une façon de pousser cette technique un peu plus loin est de demander à l'algorithme de se servir de cette information pour tenter de minimiser les coûts de classification et de test.

Cette modification, qui semble raisonnable, demande souvent des changements à l'algorithme de base assez importants. Par conséquent, élaborer un hybride qui combine une approche qui tient compte des coûts et une autre qui effectue la classification est plus complexe que les solutions présentées jusqu'à maintenant. Dans tous les cas, l'augmentation de la complexité est compensée par l'acquisition d'une méthode de classification très puissante qui peut s'adapter aux besoins de la personne nécessitant de l'information extraite de la base de données.

Une autre problématique concernant l'arbre de décision est attribuée à l'utilisation de moins en moins de données pour extraire ses règles de classification. Par conséquent, la signifiante des conditions de séparation plus loin dans les branches diminue en même temps que le nombre de données traitées à chaque noeud.

C'est le problème que Sheng et coll. [SL05] ont voulu régler. Ils ont eu l'idée de combiner un arbre de décision (C4.5) avec la méthode bayésienne naïve. Dans leur algorithme (*DTNB*), repris en détail dans la section 5, l'arbre de décision dicte quels tests doivent être effectués tandis que le classificateur bayésien effectue la classification en se basant sur l'ensemble des données disponibles.

Dans un autre ordre d'idée, Turney [Tur95] décrit une manière originale d'utiliser le principe d'évolution des algorithmes génétiques pour influencer la construction d'un arbre de décision. Cette méthode est appelée *ICET*.

Dans la méthode ICET, la fonction du coût en information est utilisée conjointement avec l'algorithme génétique qui insère des coûts supplémentaires à chaque noeud, ce qui permet de faire varier l'ordre d'utilisation des attributs dans l'arbre de décision. En introduisant un biais dans les coûts des attributs, l'arbre résultant a la chance d'avoir une meilleure vue d'ensemble du problème qui ne minimise peut-être pas le coût à court terme, mais qui permet d'avoir de meilleurs résultats à long terme en cherchant un minimum global plutôt que de s'attarder sur les minimums locaux.

4.6. Techniques diverses

Pratiquement tous les algorithmes d'apprentissage ont des paramètres qui peuvent être modifiés pour altérer la construction du modèle. Même les méthodes dites « non paramétriques », comme l'arbre de décision, possèdent certaines valeurs qui peuvent être modifiées comme la différence nécessaire entre l'erreur d'un noeud et l'erreur moyenne requise pour procéder à la construction d'un nouveau niveau.

Une méthode d'hybridation d'algorithmes employée afin de définir les paramètres d'un modèle est proposée par Remeikis et coll. [RSM] qui utilisent une

méthode similaire au DT-KBANN de Hewahi [Hew09]. Ils utilisent un arbre de décision pour d'abord étudier les données et en extraire des règles de décisions. Ces règles sont ensuite converties en une topologie pour un réseau de neurones dépendant des disjonctions et des conjonctions des règles.

Une nouvelle façon d'utiliser un algorithme génétique est donnée par Zorman et coll. [ZČGM04]. Ils se servent d'un algorithme génétique afin de trouver les meilleurs paramètres à utiliser pour la construction de leur arbre de décision.

Une autre technique d'hybridation est d'utiliser l'information découverte par un algorithme et de l'insérer sous forme d'un nouvel attribut dans le second algorithme pour l'appuyer dans sa classification. Comme le nouvel attribut créé est souvent déjà un bon indicateur de la classe à laquelle appartient une instance à classer, le second algorithme peut se concentrer sur les cas particuliers, un peu à l'image du renforcement présenté à la section 4.1.

Peddabachigari et coll. [PAGT05] discutent d'un hybride, appelé *DT-SVM*, qui mélange un arbre de décision à une machine à vecteurs de support. Leur idée consiste à construire un arbre de décision dont les feuilles sont numérotées de 1 à 2^d . Cette numérotation est ensuite convertie en un nouvel attribut (ayant 2^d valeurs) qui devrait aider la machine à vecteurs de support à bâtir son modèle puisque les instances semblables, selon l'arbre de décision, auront des valeurs identiques.

Malheureusement, les auteurs ne sont pas arrivés à des résultats très concluants contrairement à Chong et coll. [CAP05]. Leur algorithme hybride, le *DTANN*, a donné de meilleurs résultats qu'un réseau de neurones, un arbre de décision ou une machine à vecteur de support employés séparément. Le principe du DTANN est le même que celui employé par Peddabachigari et coll. [PAGT05].

Une extension viable du DTANN est d'utiliser des techniques d'analyse de groupement des données, une technique faisant partie des apprentissages non supervisés, afin de créer des groupes semblables entre eux. C'est exactement ce qu'ont fait Bose et Chen [BC09] : ces groupes peuvent être convertis en un attribut qui peut ensuite aider à la construction d'un nouveau modèle. Cette façon de faire pourrait même s'étendre à tout les types d'apprentissage supervisé. En ajoutant cette étape, il est possible d'utiliser toute la puissance des algorithmes non supervisés au service des algorithmes supervisés.

5. Étude de cas

Cette section sert à illustrer la façon d'hybrider deux algorithmes pour arriver à en créer un meilleur. À cette fin, on reprend l'idée de Sheng et coll. [SL05] pour implanter un arbre de décision qui sait prendre en considération les coûts de classification et les **coûts de test** qui représentent la pénalité d'utilisation des valeurs d'un attribut.

En pratique, il arrive que l'étude d'une base de données puisse se faire directement avec les données disponibles, comme pour l'analyse des ventes faites durant une année. Il arrive également qu'il y ait un coût relié à l'acquisition de l'information pertinente à l'étude. Si l'algorithme utilise un certain attribut, le coût de test rattaché à cet attribut doit être payé. Le but de ces coûts de test est d'obliger l'algorithme à n'utiliser que les attributs les plus pertinents.

Ce genre de classificateur est employé en médecine où l'acquisition de résultats pour chaque test occasionne des frais et où un mauvais diagnostic peut entraîner des pertes financières et humaines.

Une première façon d'utiliser une fonction de coûts pour déterminer la performance d'un arbre de décision pourrait se faire en remplaçant la fonction de segmentation (originellement basée sur l'entropie de l'information) par la fonction de coût : le but deviendrait de minimiser les coûts de mauvaise classification. En ajoutant une fonction de coût de test, l'algorithme essaie de minimiser les coûts de test et de classification en parallèle, donc d'avoir un arbre compact qui ne cherche pas à tout prix à minimiser l'erreur de classification.

Il pourrait être intéressant d'avoir une classification plus nuancée surtout dans le cadre de tests médicaux. L'arbre de décision usuel ne donne qu'une classification booléenne : l'instance fait partie d'une classe ou elle n'en fait pas partie. Il est possible d'établir la probabilité qu'une instance dans une feuille appartienne à une certaine classe en gardant les ratios d'instances d'apprentissage affichant une classe. Par exemple, une feuille ayant quatre instances lors de l'apprentissage, dont trois sont présentes dans la classe et une dans la classe , indiquerait normalement que cette feuille classe les instances s'y trouvant dans la classe . Or, il serait également possible d'indiquer qu'une instance appartenant à cette feuille a de chance d'appartenir à la classe et de chance d'appartenir à la classe . L'algorithme de classification bayésienne est un bon exemple d'algorithme qui effectue ce travail.

Malheureusement, cette technique présente quelques défauts qui peuvent être réglés grâce à l'algorithme *DTNB* de Sheng et coll. [SL05]. Cette façon de faire bâtit un arbre de décision prenant les coûts de test et de classification comme expliqué un peu plus tôt. Par contre, pour la classification, cette méthode bâtit également, en parallèle, un classificateur bayésien qui s'occupe de la classification. L'astuce réside dans le rôle joué par l'arbre de décision et le classificateur bayésien : l'arbre de décision sert à déterminer quels tests effectuer et le classificateur bayésien s'occupera de la classification en n'utilisant que certains tests, tel qu'indiqué par l'arbre.

Avant d'entrer dans les détails de cet algorithme, voici une introduction du classificateur bayésien utilisé par cette méthode d'apprentissage.

5.1. Application naïve de la théorie bayésienne

Pour la classification de données, le foreur de données dispose de fondements théoriques mathématiques qui peuvent s'incorporer à l'apprentissage comme le théorème de Bayes. Les développements en forage de données ont permis d'utiliser le théorème de Bayes afin de prendre en considération les données d'entraînement et produire la probabilité qu'une instance appartienne à une classe selon un groupe d'attributs.

Voici un rappel du théorème de Bayes. Soit et deux évènements, alors

où représente la probabilité d'un évènement sachant que l'évènement s'est réalisé. Dans cette formule, est la probabilité **à postériori** et est la probabilité **à priori**.

Soit \mathbf{x} un vecteur de dimension n où les x_i ($i = 1, \dots, n$), sont les attributs d'une instance à distribuer selon les classes de la variable dépendante c . Or, selon le théorème de Bayes :

$$P(\mathbf{x} | c) = \prod_{i=1}^n P(x_i | c)$$

où $P(x_i | c)$ est la probabilité que tous les attributs aient une certaine valeur :

ce qui mène à

$$(1) \quad P(\mathbf{x} | c) = \prod_{i=1}^n P(x_i | c)$$

Il suit, en utilisant la définition de la probabilité conditionnelle,

$$P(\mathbf{x} | c) = \frac{P(\mathbf{x}, c)}{P(c)}$$

$$P(\mathbf{x} | c) = \frac{P(\mathbf{x}, c)}{P(c)}$$

(2)

C'est ici où il faut poser l'hypothèse d'indépendance conditionnelle des événements entre eux sachant c ¹³, c'est-à-dire que $P(x_i | c, \mathbf{x}_{-i}) = P(x_i | c)$. Il suit

En appliquant cette propriété à répétition sur le résultat 2, il suit

13. À noter que l'indépendance n'implique pas l'indépendance conditionnelle, et vice versa.

et en remplaçant dans l'équation 1, il en découle

Une remarque concernant la probabilité P_{ij} qui se trouve au dénominateur : en pratique, il faut classer une instance selon la classe qui maximise P_{ij} . Or, lorsqu'il vient le temps de faire la comparaison entre P_{ij} et P_{ik} , les deux côtés de l'inéquation ont un dénominateur P_i qui peut être simplifié. Pour cette raison, les calculs sont simplifiés dès le départ. Il faut donc trouver la classe de C qui maximise :

où r_{ij} est le ratio du nombre d'instances dans la classe C_j sur le nombre d'instances au total. De la même façon, r_{ik} est le nombre d'instances de la classe C_k qui ont x_i comme attribut sur le nombre d'instances de la classe C_i . Une modification intéressante est d'utiliser des courbes normales pour calculer la probabilité P_{ij} d'un attribut numérique.

Pour classer une nouvelle instance, il suffit de trouver la classe C_j qui maximise la probabilité P_{ij} et de classer l'instance dans cette classe.

Malheureusement, en pratique il est bien rare que les attributs soient conditionnellement indépendants entre eux, d'où le nom d'application *naïve* du théorème de Bayes. Par contre, les chercheurs en apprentissage automatique ont constaté que même l'hypothèse d'indépendance conditionnelle naïvement faite, l'algorithme donnait de bons résultats. Pour ces raisons et pour ses racines dans la théorie mathématique, l'algorithme bayésien naïf est aujourd'hui très populaire, particulièrement dans le domaine de la reconnaissance du langage naturel [WF05].

5.2. Intégration de la théorie bayésienne dans l'arbre de décision

L'intégration d'un algorithme bayésien naïf dans les feuilles d'un arbre de décision poursuit un but en trois points :

- effectuer des classifications basées sur plusieurs variables à la fois ;
- prendre en compte les coûts de test et de classification ;
- décrire un modèle dynamique qui permet de minimiser les coûts de test.

Pour y arriver, voici comment l'algorithme d'arbre de décision est modifié :

- Tout d'abord, il faut spécifier les coûts de test (c'est-à-dire le coût d'acquisition des valeurs d'un attribut pour un enregistrement) de chaque attribut, en plus des coûts de mauvaise classification des instances.
- Une fois les coûts établis, un algorithme d'arbre de décision est employé sur les données. Cet arbre de décision fonctionne sur le même principe que le C4.5 à deux différences près :

- (1) La fonction de séparation des noeuds est une fonction de minimisation des coûts de mauvaise classification et de test ; cette fonction de coût est décrite plus loin.
- (2) Il n'y a pas d'élagage effectué sur l'arbre.

- De façon parallèle et indépendante de l'arbre de décision, un classificateur bayésien est créé avec l'ensemble des instances disponibles dans l'ensemble d'apprentissage.

À la fin, l'algorithme renvoie deux modèles : un arbre de décision et un classificateur bayésien.

Pour classer une nouvelle instance, l'arbre de décision est utilisé pour déterminer quels tests une nouvelle instance devrait passer afin d'être classée. Chaque test présent le long du chemin de la classification par l'arbre de décision devient un attribut pour lequel on accepte de payer le coût de test. C'est donc un processus dynamique : on fait un premier test, observe le résultat et détermine ensuite quel est le prochain test selon l'arbre de décision, jusqu'à ce que l'on parvienne à une feuille.

Une fois tous les tests d'un certain chemin effectués, l'instance peut être classée selon le modèle bayésien, en ne tenant compte que des attributs qui ont des valeurs (c'est-à-dire ceux pour lesquels un test a déjà été fait).

Bref, l'arbre de décision renseigne sur les tests (ou les attributs) qui sont utiles à la classification et le classificateur bayésien se charge de classer les instances en fonction des données disponibles.

La fonction qui sert de critère de segmentation est en fait l'espérance du coût de mauvaise classification. Soient :

- le coût d'une classification positive alors qu'elle est positive ;
- le coût d'une classification positive alors qu'elle est négative ;
- le coût d'une classification négative alors qu'elle est négative ;
- le coût d'une classification négative alors qu'elle est positive ;
- le nombre d'enregistrements positifs dans l'ensemble étudié ;
- le nombre d'enregistrements négatifs dans l'ensemble étudié.

À première vue, donner des coûts aux classifications correctes semble farfelu et est en effet souvent laissé à zéro. Par contre, il est utile de pouvoir leur donner un poids, même faible, afin de pouvoir établir des ratios entre les différentes bonnes classifications.

Le coût d'une classification positive et
le coût d'une classification négative est .

Ce coût de classification sert de critère de segmentation dans chacun des noeuds et est appliqué sur les instances de chaque noeud qu'il faut séparer.

Puisqu'il faut minimiser le coût total, la probabilité de classer un élément comme étant positif est donnée par

Ensuite, on cherche à déterminer le coût moyen d'une classification positive. Pour ce faire, on calcule l'espérance du coût de classer un élément comme étant positif :

L'espérance des coûts d'une classification négative est obtenue de façon réciproque.

Afin de pouvoir effectuer des comparaisons, on calcule l'espérance du coût de classification si l'ensemble (ou le noeud) courant n'est pas séparé. Cette espérance est définie comme étant

tandis que l'espérance du coût si l'ensemble à l'étude est séparé selon l'attribut possédant valeurs est obtenu avec

En calculant la valeur de pour tous les attributs de la base de données, il est facile de pouvoir trouver l'attribut de séparation qui minimise le coût de classification. C'est de cette façon que la fonction de coût de classification remplace le critère de segmentation de l'arbre de décision.

Donc, contrairement à l'algorithme C4.5 qui calcule le gain en information, *DTNB* calcule la réduction du coût de mauvaise classification qui est donnée par

où est le coût du test à effectuer en lien avec l'attribut .

Pour la classification d'un nouvel enregistrement, il faut passer d'abord par l'arbre de décision. Le nouvel enregistrement est introduit dans l'arbre. À chaque noeud, si l'attribut de séparation est manquant, un test est effectué. Une fois arrivé dans une feuille, l'instance (munie de ses nouvelles valeurs pour certains attributs qui étaient manquants) est soumise à la classification par le modèle bayésien. Celui-ci ignore les données manquantes et classe l'instance dans le but de minimiser le coût de mauvaise classification. L'algorithme classe un enregistrement comme étant positif si

ou négatif si l'expression est fausse.

Au niveau de la réduction des coûts, il a été montré par les auteurs que le *DTNB* est plus performant qu'un arbre de décision ou un modèle bayésien qui prennent en considération les coûts de mauvaise classification.

L'arbre utilise également l'ensemble des informations à sa disposition afin de prendre une décision quant à la classification d'un nouvel enregistrement, le but étant justement de minimiser les coûts de test pour obtenir les informations.

5.3. Et avec des données réelles...

Après avoir vu la théorie de ce nouvel hybride, il est intéressant d'observer comment il agit en pratique. La table 2 présente quelques instances sur lesquelles est appliqué le nouvel algorithme. Ensuite, il faut aussi établir des coûts de test à chacun des quatre attributs comme établis à la table 3. Finalement, il faut donner une matrice de coût de mauvaise classification telle celle présentée à la table 4. Les données présentées dans ces tableaux sont inspirées d'une batterie de tests qui pourraient être utilisée pour déterminer si oui (1) ou non (0) une personne est atteinte d'une certaine maladie, chaque test ayant un certain coût et donnant un résultat positif (1) ou négatif (0). Par conséquent, il est plus grave

de classer une personne malade comme étant saine que l'inverse. Ceci se reflète dans la matrice de coût de classification.

Instance#	Variable dépendante				
1	0	0	1	0	1
2	0	1	0	0	0
3	1	1	1	0	0
4	0	1	1	0	0
5	1	1	0	1	0
6	1	1	1	0	1
7	0	0	1	0	1
8	0	1	1	1	0
9	1	1	0	0	1
10	1	1	0	1	1

TABLE 2. Base de données utilisée pour l'application de l'algorithme DTNB.

1	5	3	2

TABLE 3. Coûts de test.

		Valeurs prédites	
		1	0
Valeurs réelles	1		
	0		

TABLE 4. Coûts de classification.

Pour effectuer la classification selon l'algorithme du DTNB, il faut d'abord créer un arbre de décision en utilisant la fonction de réduction de coût de mauvaise classification comme critère de segmentation. Afin de simplifier les choses, il faut plutôt minimiser

Par exemple, le calcul d'une première séparation par l'attribut donnerait un coût total d'environ . En effet,

et

De la même façon, il est possible de trouver $\frac{1}{2}$ et $\frac{1}{2}$. Avec ces valeurs, il suit

Sachant que le coût de test de $\frac{1}{2}$ est de 1, il suit que $\frac{1}{2}$.
 Suivant le même principe, il est possible de trouver $\frac{1}{2}$,
 et $\frac{1}{2}$.

Suivant notre algorithme, notre première segmentation se fait selon l'attribut qui minimise $\frac{1}{2}$. La suite de notre arbre se construit selon un processus similaire et se termine quand $\frac{1}{2}$, c'est-à-dire que le coût augmente si une nouvelle segmentation est créée ou s'il n'y a plus d'attribut pour faire une nouvelle segmentation.

En parallèle à la création de l'arbre, il faut créer un classificateur bayésien. Pour ce faire, il faut déterminer la probabilité qu'une instance soit positive sachant la valeur de ses attributs. En d'autres termes, on cherche :

et

Par exemple, selon les données de la table 2 on peut trouver que la probabilité à priori qu'une valeur soit positive est de $\frac{1}{2}$. Si on classe une instance avec les valeurs $\frac{1}{2}$, il suit

— — — — —

Par le même processus, on trouve que $\frac{1}{2}$. La probabilité que l'instance soit classée négative (0) étant supérieure à la probabilité que l'instance soit classée positive (1) amène le classificateur à assigner la classe négative à l'instance.

Or, l'algorithme hybride fait en sorte que l'ensemble des valeurs des attributs n'est pas disponible, tout dépendant de l'arbre de décision. Heureusement, le classificateur bayésien supporte très bien les valeurs manquantes : s'il y a des valeurs qui ne sont pas disponibles, il faut seulement ignorer les attributs correspondants.

Par exemple, si l'on classe une instance ayant comme valeur $\frac{1}{2}$, $\frac{1}{2}$,
 $\frac{1}{2}$, $\frac{1}{2}$, on obtient

— — — — —

Encore une fois, cette instance serait classée négative. Cette robustesse aux valeurs manquantes fait du classificateur bayésien un algorithme de choix pour cet hybride.

6. Conclusion

Dans cet article, il a été question de méthode d'hybridation des techniques d'apprentissage en forage de données, plus particulièrement des arbres de décision. Par exemple, un arbre de décision peut être combiné avec un algorithme bayésien naïf afin de prendre en compte la gestion des coûts de tests et de mauvaise classification.

Toutes les techniques présentées à la section 4 peuvent être utiles en situation pratique de forage de données et devraient être connues du foreur. Bien sûr, les techniques mentionnées peuvent très souvent être utilisées conjointement avec d'autres algorithmes que les arbres de décision. Notamment, beaucoup de recherches ont été faites sur les hybrides incluant un réseau de neurones.

D'une part, beaucoup de ces stratégies pourraient être appliquées de façon systématique comme le tri des attributs (section 4.2) ou les techniques d'ensembles (section 4.1).

D'autre part, il serait intéressant de faire ressortir les situations où certaines techniques sont à privilégier afin de bâtir un algorithme qui pourrait lui-même décider de ses stratégies d'hybridation. De cette façon, il serait possible d'établir des systèmes de forage de données qui, selon la nature des données, seraient en mesure de faire une batterie de tests complète et d'utiliser les informations de plusieurs de ces examens pour la construction d'un modèle efficace. Cette façon de faire pourrait unifier les cas spéciaux et transformer le forage de données en un outil utilisé de façon plus éclairée par les propriétaires de bases de données importantes.

Les informations enfouies dans les bases de données valent souvent beaucoup plus que les données qui les représentent. Il ne reste plus qu'à les forer pour en extraire toute la richesse.

Références

- [AH98] S. Anand and J. Hughes. Hybrid data mining systems : The next generation. 1394 :13–24, 1998.
- [AT07] T. Arentze and H. Timmermans. Parametric action decision trees : Incorporating continuous attribute variables into rule-based models of discrete choice. *Transportation Research Part B : Methodological*, 41(7) :772–783, 2007.
- [BAM07] A. Bouyer, B. Arasteh, and A. Movaghar. A new Hybrid Model using Case-Based Reasoning and Decision Tree Methods for improving Speedup and Accuracy. In *Iadis International Conference Applied Computing*, pages 978–972, 2007.
- [BC09] I. Bose and X. Chen. Hybrid models using unsupervised clustering for prediction of customer churn. *Proceedings of the International MultiConference of Engineers and Computer Scientists*, 1, 2009.
- [BFOS84] L. Breiman, J.H. Friedman, R.A. Olshen, and C.J. Stone. Classification and regression trees. *Wadsworth, Belmont*, 1984.
- [BHV 95] J. Bala, J. Huang, H. Vafaie, K. DeJong, and H. Wechsler. Hybrid learning using genetic algorithms and decision trees for pattern classification. In *International Joint Conference on Artificial Intelligence*, volume 14, pages 719–724, 1995.
- [Bre96] L. Breiman. Bagging predictors. *Machine learning*, 24(2) :123–140, 1996.

- [CAP05] M. Chong, A. Abraham, and M. Paprzycki. Traffic accident analysis using machine learning paradigms. *Special Issue : Computational Intelligence in Data mining Guest Editors : Janos Abonyi*, 29 :89, 2005.
- [CF00] D.R. Carvalho and A.A. Freitas. A hybrid decision tree/genetic algorithm for coping with the problem of small disjuncts in data mining. In *Proc. Genetic and Evolutionary Computation Conf (GECCO-2000)*, pages 1061–1068, 2000.
- [CF04] D.R. Carvalho and A.A. Freitas. A hybrid decision tree/genetic algorithm method for data mining. *Information Sciences*, 163(1-3) :13–35, 2004.
- [FS95] Y. Freund and R. Schapire. A decision-theoretic generalization of on-line learning and an application to boosting. In *Computational learning theory*, pages 23–37. Springer, 1995.
- [FSBA09] S. Fong, Y.W. Si, and R.P. Biuk-Aghai. Applying a hybrid model of neural network and decision tree classifier for predicting university admission. In *Information, Communications and Signal Processing, 2009. ICICS 2009. 7th International Conference on*, pages 1–5. IEEE, 2009.
- [FWI 98] E. Frank, Y. Wang, S. Inglis, G. Holmes, and I.H. Witten. Using model trees for classification. *Machine Learning*, 32(1) :63–76, 1998.
- [GK95] S. Goonatilake and S. Khebbal. *Intelligent hybrid systems*. John Wiley & Sons, 1995.
- [Hew09] N.M. Hewahi. A Hybrid Architecture for a Decision Making System. *Journal of Artificial Intelligence*, 2(2) :73–80, 2009.
- [HK06] J. Han and M. Kamber. *Data Mining : Concepts and Techniques*. Morgan Kaufmann, deuxième édition, 2006.
- [KB03] Y. Kornienko and A. Borisov. Investigation of a hybrid algorithm for decision tree generation. In *Intelligent Data Acquisition and Advanced Computing Systems : Technology and Applications, 2003. Proceedings of the Second IEEE International Workshop on*, pages 63–68. IEEE, 2003.
- [KG10] M.A. Kumar and M. Gopal. A hybrid SVM based decision tree. *Pattern Recognition*, 43(12) :3977–3987, 2010.
- [Koh96] R. Kohavi. Scaling up the accuracy of naive bayes classifiers : a decision-tree hybrid. *KDD-96 Proceedings*, 1996.
- [KR10] M. Kissi and M. Ramdani. A hybrid decision trees-adaptive neuro-fuzzy inference system in prediction of anti-HIV molecules. *Expert Systems With Applications*, 2010.
- [LHF03] N. Landwehr, M. Hall, and E. Frank. Logistic model trees. *Machine Learning : ECML 2003*, pages 241–252, 2003.
- [NMS10] B.B. Nair, VP Mohandas, and NR Sakthivel. A Decision Tree-Rough Set Hybrid System for Stock Market Trend Prediction. *International Journal of Computer Applications*, 6(9) :1–6, 2010.
- [Nou] H.F.T. Noubi. Feature Selection & Hybrid Decision Tree/Genetic Algorithm for Cancer Diagnosis based on Mass Spectrometry Proteomics.
- [PAGT05] S. Peddabachigari, A. Abraham, C. Grosan, and J. Thomas. Modeling intrusion detection system using hybrid intelligent systems. *Journal of Network and Computer Applications*, 2005.
- [PN03] H. Pistori and J.J. Neto. Decision tree induction using adaptive FSA. *CLEI Electron. J.*, 6(1), 2003.
- [Qui86] J.R. Quinlan. Induction of decision trees. *Machine learning*, 1(1) :81–106, 1986.
- [Qui93] J.R. Quinlan. *C4. 5 : programs for machine learning*. Morgan Kaufmann, 1993.
- [RN94] S.J. Russell and P. Norvig. *Artificial Intelligence : A Modern Approach*. Prentice Hall, 1994.

- [RSM] N. Remeikis, I. Skučas, and V. Melninkaitė. Hybrid Machine Learning Approach for Text Categorization. *International Journal of Computational Intelligence*, 1(1).
- [SCWH05] G. Stein, B. Chen, A.S. Wu, and K.A. Hua. Decision tree classifier for network intrusion detection with GA-based feature selection. In *Proceedings of the 43rd annual Southeast regional conference-Volume 2*, pages 136–141. ACM, 2005.
- [SL05] S. Sheng and C. Ling. Hybrid cost-sensitive decision tree. *Knowledge Discovery in Databases*, 2005.
- [SPW01] A. Seewald, J. Petrak, and G. Widmer. Hybrid decision tree learners with alternative leaf classifiers : An empirical study. *audiology*, 2001.
- [Tur95] P. Turney. Cost-sensitive classification : Empirical evaluation of a hybrid genetic decision tree induction algorithm. *Journal of Artificial Intelligence Research (JAIR)*, 2, 1995.
- [TW09] CF Tsai and SP Wang. Stock price forecasting by hybrid machine learning techniques. *Proceedings of the International MultiConference of Engineers and Computer Scientists*, 1 :755–60, 2009.
- [WF05] I. Witten and E. Frank. *Data Mining Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, deuxième édition, 2005.
- [WZX10] L. Wang, X. Zang, and P. Xu. Induction of a novel hybrid decision forest model based on information theory. *Journal of Software*, 5(11) :1195–1199, November 2010.
- [ZC02] Z.H. Zhou and Z.Q. Chen. Hybrid decision tree. *Knowledge-based systems*, 15(8) :515–528, 2002.
- [ZČGM04] M. Zorman, B. Černohorski, G. Goršek, and O. Milan. Hybrid evolutionary built decision trees for prediction of perspective cross-country skiers. In *Proceedings of the 4th WSEAS International Conference on Applied Informatics and Communications*, pages 1–6. World Scientific and Engineering Academy and Society (WSEAS), 2004.

ADAM SALVAIL-BÉRARD, DÉPARTEMENT D'INFORMATIQUE, UNIVERSITÉ DE SHERBROOKE
 Courriel: Adam.Salvail-Berard@USherbrooke.ca

TRIANGULATIONS, CARQUOIS ET THÉORÈME DE PTOLÉMÉE

GUILLAUME DOUVILLE

1. Introduction

Les algèbres amassées sont des q -algèbres commutatives, ou anneaux commutatifs, de polynômes à coefficients entiers, qui sont munies d’une structure combinatoire. Cette classe d’algèbres a été définie par Sergey Fomin et Andreï Zelevinsky en 2002, voir [FZ02]. L’objectif était d’étudier le caractère combinatoire de la positivité totale et des bases canoniques duales dans les groupes de Lie. Les algèbres amassées sont engendrées par les variables amassées, obtenues par des mutations de carquois (on réfère le lecteur à [Bou11] pour les définitions de base sur les algèbres amassées). Les algèbres amassées ont par la suite trouvé des applications et interprétations dans plusieurs domaines mathématiques et physiques, par exemple les systèmes intégrables, la géométrie de Poisson et la thermodynamique.

L’objectif de cet article est de fournir une méthode de calcul des variables amassées. En effet, nous verrons qu’il est possible d’associer à une triangulation d’une surface marquée un carquois. Pour ce faire, il faudra commencer par quelques définitions et résultats de topologie des surfaces. Par la suite, en définissant une mutation de triangulations analogue à la mutation de carquois, on obtient le calcul des variables amassées.

Ensuite, nous verrons les pavages. Plusieurs résultats importants ont été obtenus en passant par le concept de frises (voir [Cha11] pour une introduction aux frises). Pourtant, les frises, pour les carquois les plus simples, ceux de type 2 , ne sont que des pavages, qui ont été étudiés pour la première fois par John H. Conway et Harold S. Coxeter en 1973, voir [CC73]. Nous verrons dans cet article qu’il existe une bijection entre les triangulations de polygones et les pavages.

Mes remerciements à David Boulet-St-Jacques et à Myriam Chabot pour l’aide apportée lors de la rédaction de cet article. Mes remerciements aussi à tous ceux qui ont donné un séminaire d’algèbres amassées cet été, l’article n’aurait pu voir le jour sans vous. Un remerciement particulier à Ibrahim Assem et à Thomas Brüstle pour m’avoir offert ce stage, ainsi que pour la supervision et les conseils. Finalement, un remerciement à Thomas Brüstle et au CRSNG pour le financement de ce stage.

Finalement, l'article terminera avec une preuve du théorème de Ptolémée, un théorème datant de la Grèce Antique qui s'est trouvé être lié avec la méthode de calcul des variables amassées par les mutations de triangulations. La preuve présentée requiert des notions de géométrie projective et la définition des transformations de Möbius, qui ont la propriété de conserver le cercle. Nous terminerons par quelques propriétés de ces transformations en géométrie hyperbolique.

2. Surfaces, triangulations et carquois associés

2.1. Surfaces

Les triangulations de surfaces orientées sans ponction permettent la construction d'algèbres amassées. En effet, nous verrons plus loin dans cet article qu'à toute triangulation est associée un unique carquois.

Tout d'abord, voyons quelques définitions préliminaires qui mèneront aux surfaces : les espaces topologiques et les homéomorphismes.

Définition 2.1.

Un **espace topologique** est un couple (X, τ) , où X est un ensemble et τ un sous-ensemble de l'ensemble des parties de X , appelé ensemble des **ouverts** de X . L'ensemble des ouverts τ vérifie les propriétés suivantes :

- (1) \emptyset et l'ensemble vide sont inclus dans τ ;
- (2) Toute réunion d'ouverts de X est un ouvert de X ;
- (3) Toute intersection finie d'ouverts de X est un ouvert de X .

Tout ensemble qui est le complément d'un ensemble ouvert est dit **fermé**.

De plus, un espace topologique est un **espace métrique** s'il est muni d'une distance. Par exemple, \mathbb{R}^n est un espace métrique muni de la distance euclidienne usuelle, que nous rappelons ici.

Soient x, y deux points de \mathbb{R}^n . On note $d(x, y)$ la distance entre ces deux points, donnée par

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}.$$

Rappelons la définition d'une fonction continue sur un espace topologique.

Définition 2.2.

Soit X, Y des espaces topologiques. Une fonction $f: X \rightarrow Y$ est dite **continue** si et seulement si pour tout ouvert V de Y , la préimage $f^{-1}(V)$

est un ouvert de X .

Nous passons maintenant à la définition d'homéomorphisme.

Définition 2.3.

Deux espaces topologiques X et Y sont **homéomorphes** s'il existe une bijection continue $f: X \rightarrow Y$ telle que f^{-1} soit aussi continue.

L'application f est alors appelée un **homéomorphisme**.

Définition 2.4.

Une **surface** est un espace topologique où le voisinage de tout point est homéomorphe au plan.

Dans cet article, nous nous intéresserons plus particulièrement aux surfaces compactes, connexes et orientables de genre nul. Nous verrons donc les définitions de ces concepts. De plus, nous présentons la définition d'une courbe, celle-ci étant nécessaire à la définition du genre d'une surface.

Définition 2.5.

Soit S une surface. Une **courbe** est l'image d'une fonction continue $\gamma : [a, b] \rightarrow S$:

Définition 2.6.

On dit qu'une surface S dans \mathbb{R}^n est **compacte** si elle est à la fois bornée, c'est-à-dire qu'il existe un $M > 0$ et un $\epsilon > 0$ tels que pour tout $x \in S$, on ait $\|x\| \leq M$, et fermée.

Définition 2.7.

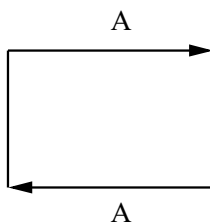
On dit qu'une surface S est **connexe** si les seuls sous-ensembles à la fois ouverts et fermés de S sont l'ensemble vide et S elle-même.

Pour définir le concept de surface orientable, il est utile de rappeler ce qu'est un ruban de Möbius.

Définition 2.8.

Un **ruban de Möbius** est un rectangle $[0, 1] \times [0, 1]$ ayant deux côtés opposés identifiés par la relation $(x, 0) \sim (x, 1)$ pour tout $x \in [0, 1]$.

On peut représenter le ruban de Möbius comme suit :

**Définition 2.9.**

On dit qu'une surface S est **orientable** si aucun sous-ensemble de S n'est homéomorphe au ruban de Möbius.

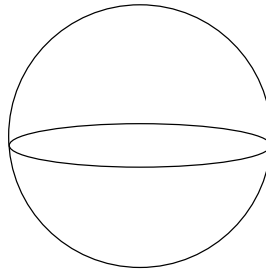
Définition 2.10.

Le **genre** d'une surface orientable est le nombre maximal de courbes fermées que l'on peut tracer sur la surface sans couper cette dernière en plusieurs parties connexes.

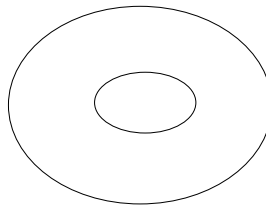
Exemple.

Voici quelques surfaces compactes, orientables et connexes :

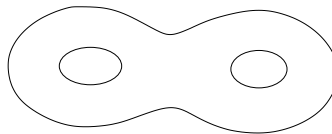
- (1) La sphère S^2 est une surface de genre 0.



(2) Le tore est une surface de genre 1.



(3) L'union de deux tores est une surface de genre 2.



(4) De même, l'union de g tores est une surface de genre g .

Nous en arrivons à un théorème qui permet de simplifier l'étude des surfaces d'intérêt pour les algèbres amassées, c'est-à-dire les surfaces compactes, orientables et connexes.

Théorème 2.11 ([Mas89]).

Toute surface compacte, orientable et connexe dans \mathbb{R}^3 est homéomorphe soit à \mathbb{S}^2 , soit à la somme de g tores.

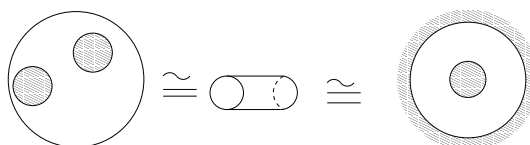
Dans le cas le plus simple, on souhaite considérer la triangulation de polygones représentés sur un disque. Il est donc nécessaire de considérer l'ajout de frontières à nos surfaces, de façon à obtenir par homéomorphisme des figures sur le plan.

Définition 2.12.

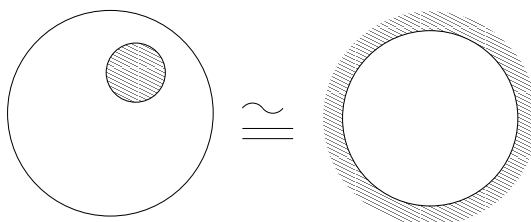
Soit S une surface et D_1, \dots, D_n des disques disjoints de frontières circulaires dans S . On dit que, à homéomorphisme près, (S, D_1, \dots, D_n) est une **surface avec frontière**.

Exemples.

(1) La surface (\mathbb{S}^2, D_1, D_2) est homéomorphe à un cylindre.



(2) La surface S est homéomorphe à un disque.



Nous commençons l'étude des triangulations des surfaces marquées sans ponction.

Définition 2.13.

Soit S une surface et P un ensemble fini de points tels que $P \cap \partial S = \emptyset$ ou $P \cap \partial S \neq \emptyset$. La **surface marquée** (S, P) est la paire (S, P) . On dit de plus que la surface est **sans ponction** si $P = \emptyset$.

Définition 2.14.

Soit (S, P) une surface marquée. Soient γ_1, γ_2 deux fonctions

et

On pose $\gamma_1 \sim \gamma_2$ et $\gamma_1 \sim \gamma_2$. Une **homotopie** est une fonction continue

telle que $\gamma_1(t) = \gamma_2(t)$ et $\gamma_1(t) = \gamma_2(t)$, pour tout

Les courbes γ_1 et γ_2 sont alors dites **homotopes**.

Il est facile de prouver que l'homotopie entre des courbes est une relation d'équivalence. On appelle **classes d'homotopie** les différentes classes d'équivalences formées par cette relation.

Définition 2.15.

Soit (S, P) une surface marquée. Un **arc** dans (S, P) est une classe d'homotopie de courbes γ qui ne s'auto-intersectent pas et telles que $\gamma(0) \in P$ et $\gamma(1) \in P$, c'est-à-dire que ces courbes commencent et se terminent en des points appartenant à P .

La triangulation est l'opération qui permet, à l'aide d'arcs, de transformer une surface marquée en une union finie de surfaces triangulaires. Les triangulations usuelles de polygones sont en fait la triangulation d'une surface sphérique à une frontière, homéomorphe à un disque dans le plan \mathbb{R}^2 . Ainsi, trianguler un rectangle revient à en tracer une diagonale.

Définition 2.16.

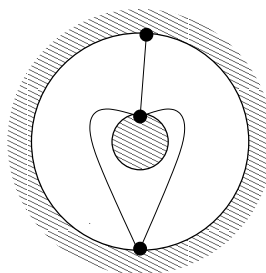
Soient (S, P) une surface avec frontières et P un ensemble fini de points marqués appartenant à P . Une **triangulation** est un ensemble maximal d'arcs qui ne s'intersectent qu'en P .

Exemples.

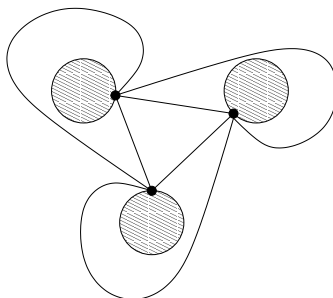
- (1) Soit \mathcal{D} un disque et S un ensemble de n points sur $\partial\mathcal{D}$. Alors $\mathcal{D} \setminus S$ est homéomorphe à un pentagone. Une triangulation possible est T , telle qu'indiquée sur la figure suivante.

On appellera les arcs a_1 et a_2 arcs internes tandis que a_3 et a_4 seront les arcs de frontières.

- (2) Soit \mathcal{D} un disque, et S un ensemble de n points sur $\partial\mathcal{D}$. Une triangulation possible est :



- (3) Soient $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3$ trois disques, et S un ensemble de n points sur $\partial\mathcal{D}$. Une triangulation possible est alors :



Dans ce qui suit, on note $\#A$ la cardinalité d'un ensemble A .

Théorème 2.17 ([FST08]).

Soient S une surface marquée sans ponction, g son genre et n le nombre de disques retirés. Alors le nombre $T(S, n)$ d'arcs internes d'une triangulation de $S \setminus \bigcup_{i=1}^n \mathcal{D}_i$ est

Corollaire 2.18.

Soient T et T' deux triangulations de P . Alors $\#T = \#T'$, c'est-à-dire que le nombre d'arcs internes de T et de T' sont égaux.

Démonstration. Cela suit directement du théorème précédent.

Nous verrons maintenant le lien entre triangulations de surfaces marquées et algèbres amassées. Comme il a été dit précédemment, à toute triangulation est associé un unique carquois.

Définition 2.19.

Soit T une triangulation d'une surface marquée sans ponction. Le carquois associé à la triangulation, noté $Q(T)$, est ainsi défini :

- L'ensemble de ses points est l'ensemble des arcs internes de T ;
- L'ensemble de ses flèches est l'ensemble des flèches (a, b) telles que a et b est un prédécesseur de c , selon l'orientation horaire.

Exemple.

Considérons le cas du pentagone. On a alors que

Le carquois associé à une telle triangulation est

Comme tout carquois peut être muté en un nouveau carquois, et que chacun de ces carquois est associé à une triangulation, il serait intéressant de connaître une façon de muter les triangulations directement.

Définition 2.20.

Soient T une triangulation d'une surface marquée et a un arc interne de T . Considérons le quadrilatère Q dont a est une diagonale. On définit le **flip** de Q par Q' où a' est la seconde diagonale de Q .

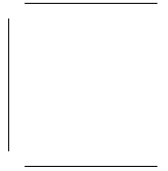


Théorème 2.21 ([Hat91]).

Soient T et T' deux triangulations de S . Alors il existe une suite de flips permettant de transformer T en T' .

Lemme 2.22.

Le carquois associé au flip en T d'une triangulation T est le même que le carquois obtenu par la mutation en T' du carquois associé à la triangulation T . En particulier, le diagramme suivant est commutatif :



Démonstration. Soit T une triangulation d'une surface marquée sans ponction. Soit α un arc interne. On peut former un quadrilatère dont α est la diagonale et les cotés des arcs de T .

Soit \mathcal{Q} le carquois associé à la triangulation T . On peut considérer, sans perte de généralité, \mathcal{Q} comme étant le sous-carquois associé aux arcs internes ci-dessus, car la mutation est un processus local.

On voit que \mathcal{Q} et \mathcal{Q}' sont tous deux égaux au carquois suivant :



Soit T une triangulation de la surface marquée sans ponction S . Nous savons que si T a n arcs internes, alors il lui est associé un carquois \mathcal{Q} ayant n points. Soit \mathcal{A} un ensemble de n indéterminées associées aux points de \mathcal{Q} . Alors la triangulation T induit une algèbre amassée \mathcal{A} , comme vu dans [Bou11]. Il suit du théorème 2.21 et du lemme 2.22 que cette algèbre amassée ne dépend pas de la triangulation T , mais uniquement de la surface marquée S .

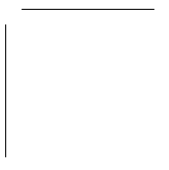
Définition 2.23.

Soit S une surface marquée sans ponction et T une triangulation de S . Alors, l'algèbre amassée \mathcal{A} associée à T est l'algèbre \mathcal{A}_T .

Nous savons que les points de \mathcal{Q} correspondent aux arcs internes de T . Soit ainsi α la variable amassée correspondant à l'arc α de T . Alors, la formule

d'échange de Fomin et Zelevinsky (voir [Bou11]) s'écrit

pour le quadrilatère

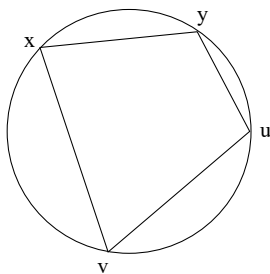


où flip note le flip de σ .

Il s'agit de l'équivalent de la relation de Ptolémée. Cette relation sera approfondie dans la section 3.4. Néanmoins, en voici l'énoncé :

Théorème de Ptolémée.

Soient σ et τ deux ordres distincts et concentriques. Alors, le produit des diagonales d'un quadrilatère inscriptible est égal à la somme des produits des paires de côtés opposés :



2.2. Pavage et triangulation

Définition 2.24.

Un **pavage** d'ordre n consiste en n lignes, la première et la dernière étant constituées de 1, telles que pour toute maille

on ait la relation

Exemple.

Voici un pavage d'ordre 7.

Théorème 2.25 ([CC73]).

Tout pavage d'ordre n est périodique de période n .

Nous verrons maintenant qu'il y a un lien entre triangulations d'un polygone et pavages. En effet, pour tout pavage, la deuxième ligne de celui-ci correspondra au nombre de triangles adjacents à chaque sommet.

Notation 2.26.

Soit T une triangulation d'un polygone à n côtés, on note $P(T)$ le pavage (au sens de [CC73]) dont la deuxième ligne est donnée par le nombre de triangles adjacents à chaque sommet de ce polygone.

Exemple.

Considérons la triangulation suivante de l'hexagone. Les nombres associés aux sommets représentent le nombre de triangles adjacents à ce sommet.



Le pavage associé à cette triangulation de l'hexagone est alors :

Théorème 2.27 ([CC73]).

Soient \mathcal{T}_n l'ensemble des triangulations du polygone à n côtés, où $n \geq 3$, et

l'ensemble des pavages d'ordre n , où $n \geq 1$. Il existe une bijection

3. Autour du théorème de Ptolémée

3.1. Géométrie projective

Le but principal de cette section est d'arriver à une preuve du théorème de Ptolémée en utilisant des outils de géométrie projective. Nous profiterons aussi de cet article pour présenter quelques résultats intéressants liés à cette géométrie.

En premier lieu, on souhaite définir l'espace complexe projectif \mathbb{CP}^1 .

Définition 3.1.

L'espace complexe projectif \mathbb{CP}^1 est l'ensemble des droites complexes contenant l'origine, c'est-à-dire que $\mathbb{CP}^1 = \{L \mid L \text{ droite complexe contenant } 0\}$, où L si et seulement s'il existe un $z \in \mathbb{C}$ tel que $L = \{t(1, z)\} \subset \mathbb{C}^2$.

Nous notons ℓ_z la droite passant par le point $(1, z)$ et l'origine.

Nous présentons une bijection entre l'espace complexe projectif et un espace plus intuitif. Pour y arriver, nous devons accepter d'inclure un point infini au plan complexe.

Définition 3.2.

Soient \mathbb{C} et \mathbb{CP}^1 . Considérons l'ajout d'un point appelé infini, noté ∞ , au plan complexe \mathbb{C} , défini de façon à respecter les énoncés suivants :



Voici la fonction bijective entre les droites de \mathbb{CP}^1 et le plan complexe muni du point infini ∞ , un plan dont les extrémités infinies seraient identifiées en un point.

Théorème 3.3 ([And07], [Tot02]).

Il existe une bijection entre \mathbb{CP}^1 et $\mathbb{C} \cup \{\infty\}$, définie ainsi :

$$z \mapsto \begin{cases} z & \text{si } z \neq \infty \\ \infty & \text{si } z = \infty \end{cases}$$

Maintenant que nous avons défini \mathcal{M} , il est possible de définir des fonctions intéressantes sur cet ensemble appelées les transformations de Möbius. En effet, ces transformations auront plusieurs propriétés utiles, entre autres, à la preuve que l'on souhaite présenter du théorème de Ptolémée. De plus, nous verrons en ouverture des propriétés des transformations de Möbius utiles en géométrie hyperbolique

3.2. Transformation de Möbius

Définition 3.4.

Soient $a, b, c, d \in \mathbb{C}$. Une **transformation de Möbius** est une fonction f telle que :

avec $ad - bc \neq 0$.

Pour les preuves subséquentes, il sera souvent plus aisé de travailler avec des fonctions plus simples que les transformations de Möbius. C'est pour cette raison que nous présentons une composition de fonctions égale aux transformations de Möbius.

Proposition 3.5.

Soit a un complexe différent de 0. Toute transformation de Möbius est une composition des fonctions complexes suivantes :

$z \mapsto z + a$ (translation)

$z \mapsto \lambda z$ (homothétie)

$z \mapsto -\frac{1}{\bar{z}}$ (inversion)

Démonstration. Soit f une transformation de Möbius. (On a que $f(z) = \frac{az + b}{cz + d}$).

Si $c \neq 0$, alors $f(z) = \frac{a}{c} + \frac{b - \frac{a}{c}d}{cz + d}$. Supposons donc que $c = 0$.

On obtient par division que

Il est alors facile de vérifier que

Voyons maintenant les propriétés de ces transformations. Premièrement, celles-ci sont non seulement inversibles, mais leurs inverses sont aussi des transformations de Möbius. De plus, les transformations de Möbius sont stables pour l'opération de composition. Comme nous le verrons, ces deux propriétés sont centrales à la preuve du théorème de Ptolémée.

Proposition 3.6.

L'ensemble des transformation de Möbius est un groupe pour la composition.

Démonstration. Soient T_1 et T_2 des transformations de Möbius définies par :

$$T_1(z) = \frac{az + b}{cz + d},$$

$$T_2(z) = \frac{az + b}{cz + d},$$

avec $ad - bc \neq 0$ et $ad - bc \neq 0$.

Premièrement, la fonction identité $T(z) = z$ est une transformation de Möbius. De plus, on obtient aisément que :

$$T_1 \circ T_2(z) = \frac{a_1(a_2z + b_2) + b_1}{c_1(a_2z + b_2) + d_1},$$

Finalement, on a que :

$$T_1 \circ T_2(z) = \frac{a_1a_2 + b_1c_2}{a_1c_2 + b_1d_2}z + \frac{a_1b_2 + b_1d_2}{a_1c_2 + b_1d_2},$$

ce qui est une transformation de Möbius.

Il importe maintenant de généraliser la notion de cercle dans $\mathbb{C} \cup \{\infty\}$, afin de présenter une propriété de conservation des cercles par les transformations de Möbius.

Définition 3.7.

Si γ est une droite passant par l'infini dans $\mathbb{C} \cup \{\infty\}$, c'est-à-dire l'union d'une droite dans \mathbb{C} et du point ∞ , ou un cercle dans \mathbb{C} , alors γ est un **cercle généralisé**.

Théorème 3.8 ([Sch79]).

Si T est une transformation de Möbius et γ un cercle généralisé, alors $T(\gamma)$ est un cercle généralisé.

Cette propriété nous rapproche du but, car, on le rappelle, la relation de Ptolémée n'est vraie que pour un quadrilatère inscrit dans un cercle. Il est donc essentiel que les transformations de Möbius conservent le cercle généralisé.

Il reste cependant un dernier obstacle avant de présenter la preuve du Théorème de Ptolémée. Il est nécessaire de définir un rapport entre les côtés d'un quadrilatère. Rapport qui serait, idéalement, en lien avec les transformations de Möbius. À cette fin, nous présentons la définition du birapport.

3.3. Birapport

Définition 3.9.

Soient z_1, z_2, z_3, z_4 , tous distincts. Le **birapport** de z_1, z_2, z_3, z_4 , noté $[z_1, z_2, z_3, z_4]$, ou parfois (z_1, z_2, z_3, z_4) , est donné par :

$$[z_1, z_2, z_3, z_4] = \frac{(z_1 - z_3)(z_2 - z_4)}{(z_1 - z_4)(z_2 - z_3)},$$

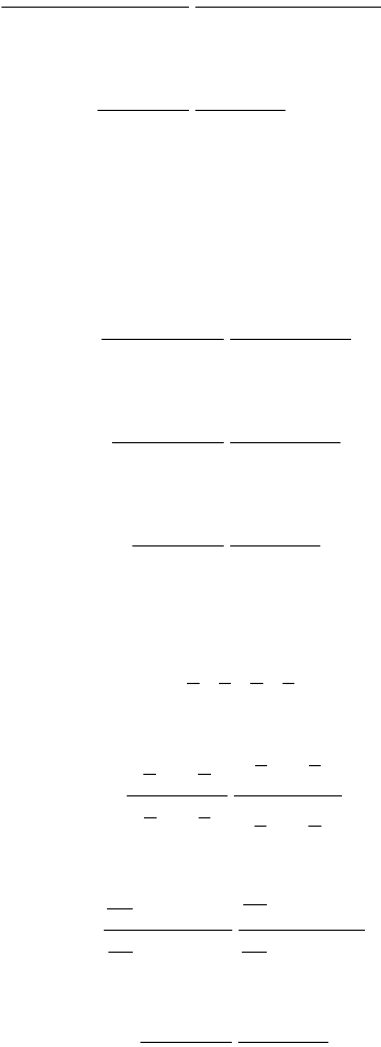
Voyons maintenant un premier lien entre les transformations de Möbius et le birapport.

Théorème 3.10.

Soient a, b, c, d, e, f , tous distincts. Si φ est une transformation de Möbius, alors on a que :

Démonstration. Soit φ une transformation de Möbius. Soient a, b, c, d, e, f distincts dans $\mathbb{C} \cup \{\infty\}$.

Comme φ est une composition de $z \mapsto z + t$ et $z \mapsto \frac{1}{z}$, il suffit de prouver que chacune de ces fonctions conserve le birapport.



Voici la propriété centrale qui montre l'intérêt des transformations de Möbius et du birapport. Une fonction qui envoie tout point vers son birapport avec trois points arbitraires, mais distincts, a la propriété d'être un birapport.

Proposition 3.11.

Soient z_1, z_2, z_3 , tous distincts. Considérons la fonction complexe suivante :

Alors, $f(z)$ est une transformation de Möbius.

Démonstration. Soient z_1, z_2, z_3 tous distincts.

Il suffit de montrer que $f(z)$ s'écrit sous la forme $\frac{az+b}{cz+d}$ avec

$ad-bc \neq 0$ et z_1, z_2, z_3 distincts.

Or,

$$f(z) = \frac{(z - z_1)(z_2 - z_3)}{(z - z_3)(z_2 - z_1)}$$

où z_1, z_2, z_3 sont distincts, et $ad-bc \neq 0$.

De plus, z_1, z_2, z_3 et

z_1, z_2, z_3 , car z_1, z_2, z_3 sont distincts en vertu de l'hypothèse.

Remarquons que la transformation de Möbius $f(z)$ est munie de la propriété essentielle de fixer l'image de z_1, z_2, z_3 . En effet, on a

pour tous z .

Grâce à la proposition et à la remarque précédentes, il est possible d'établir le corollaire suivant.

Corollaire 3.12.

Pour tous z_1, z_2, z_3 distincts et pour tous w_1, w_2, w_3 distincts, il existe une transformation de Möbius f telle que

Démonstration. Cela suit directement de la proposition 3.11. En effet, il suffit de considérer les fonctions $f_1(z)$ et $f_2(z)$. Il s'ensuit que $f = f_2 \circ f_1^{-1}$ est telle que :

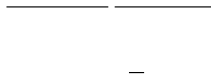
Ce corollaire simplifiera grandement les preuves subséquentes, puisqu'il sera possible de considérer seulement un triplet choisi de points (à savoir x, y, z et u) sans perte de généralité.

Pour l'instant, nous avons considéré le choix de points arbitraires distincts. Nous prouvons maintenant que le choix de points appartenant à un même cercle (généralisé) mènera à l'obtention d'un birapport réel malgré le fait qu'a priori, le birapport soit un nombre appartenant à \mathbb{C} .

Théorème 3.13.

Soient x, y, z, u tous distincts. Alors le birapport $\frac{(x-y)(z-u)}{(x-u)(z-y)}$ appartient à \mathbb{R} si et seulement si x, y, z, u appartiennent à un même cercle généralisé.

Démonstration. Soit $\frac{(x-y)(z-u)}{(x-u)(z-y)}$. En vertu du corollaire précédent, on peut considérer sans perte de généralité que x, y, z, u sont des points réels.



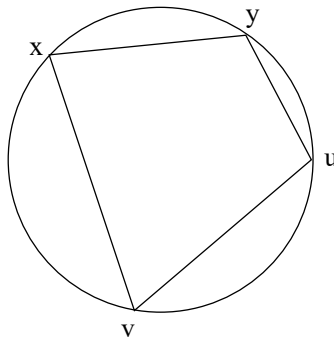
et cette dernière expression est réelle si et seulement si $\frac{(x-y)(z-u)}{(x-u)(z-y)}$ appartient aux réels si et seulement si x, y, z, u se retrouvent sur l'axe réel dans \mathbb{R} , donc sur le même cercle généralisé dans \mathbb{C} .

3.4. Théorème de Ptolémée

Comme nous l'avons vu précédemment, la nouvelle variable obtenue d'un flip est déterminée par une formule rappelant le théorème de Ptolémée. Voici maintenant la preuve de ce théorème :

Théorème de Ptolémée.

Soient x, y, z, u, v distincts et sur un même cercle. Alors,



Démonstration. Considérons une transformation de Möbius telle que

On a alors que :



L'avant-dernière égalité suivant du fait que et la dernière venant par application de .
Ainsi, on a que , d'où on peut conclure à l'égalité des distances :

3.5. Quelques propriétés des transformations de Möbius en géométrie hyperbolique

Cette section s'éloigne quelque peu de l'étude des algèbres amassées, néanmoins elle est en lien direct avec les outils développés dans cet article en vue d'obtenir une preuve du théorème de Ptolémée. Nous présentons des liens entre transformations de Möbius et géométrie hyperbolique.

Notation 3.14.

On note le demi-plan complexe supérieur muni du point à l'infini, c'est-à-dire l'ensemble .

Commençons par définir la distance la plus courte entre deux points, en quelque sorte le segment de droite de la géométrie hyperbolique, puis l'angle entre deux courbes.

Définition 3.15.

Soient , soit l'ensemble des courbes entre et , . Alors, la **distance hyperbolique** entre et est donnée par



Remarque. $\|\cdot\|$ note la norme euclidienne usuelle d'une courbe paramétrisée dans \mathbb{R}^2 .

Nous verrons plus loin une façon concrète de calculer cette distance, car l'ensemble des longueurs des courbes entre 1^- et 1^+ est de cardinalité infinie. La preuve de cette méthode peut être obtenue au moyen des transformations de Möbius.

Théorème 3.16.

Soit T une transformation de Möbius telle que :

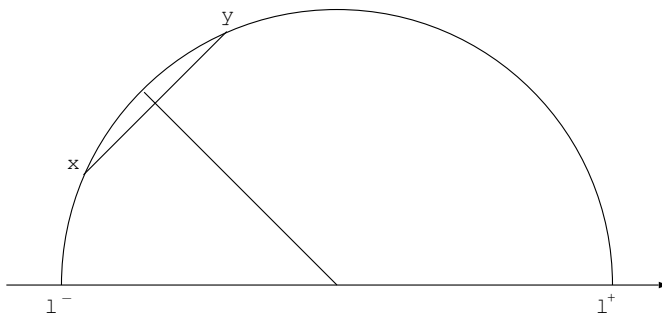
où α, β sont réels.

Alors, T préserve le demi-plan supérieur \mathbb{H} .

Notation 3.17.

Soient 1^- et 1^+ . On note x et y les points d'intersection entre l'axe réel et le demi cercle passant par 1^- et 1^+ et centré sur l'axe réel.

On convient de choisir x et y de façon à avoir $x < y$.

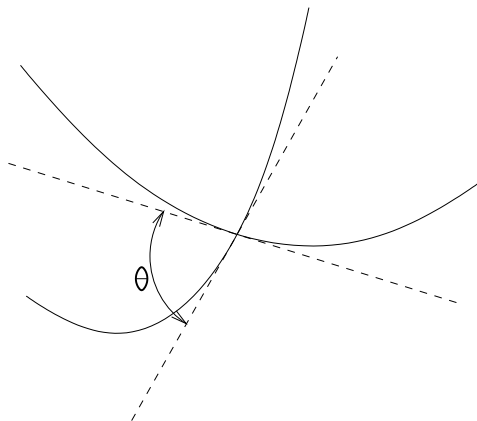


Proposition 3.18 ([And07]).

Soient T et U . Alors, $\angle(T, U)$ est donnée par la longueur de l'arc de cercle entre 1^- et 1^+ défini en 3.17.

Définition 3.19.

L'angle entre deux courbes dans \mathbb{H} est défini comme étant l'angle entre ces deux courbes dans \mathbb{H} , c'est-à-dire l'angle entre les droites tangentes au point d'intersection des deux courbes.



Proposition 3.20 ([And07]).

Soient T une transformation de Möbius et γ, δ deux courbes dans \mathbb{H}^2 . Alors, T préserve l'angle entre γ et δ .

Références

- [And07] J.W. Anderson. *Hyperbolic Geometry*, 2^e éd., Springer (2007), 276 p.
- [Bou11] D. Boulet-St-Jacques. *Les algèbres amassées : Définitions de base et résultats* (2011), soumis pour publication.
- [Cha11] M. Chabot. *Fonctions de frises et algèbres amassées* (2011), soumis pour publication.
- [CC73] J.H. Conway et H.S.M. Coxeter. *Triangulated Polygons and Frieze Patterns*, The Mathematical Gazette 57, no. 400 (1973) 87-94.
- [FST08] S. Fomin, M. Shapiro et D. Thurston. *Cluster Algebras and Triangulated Surfaces*, Part I : Cluster complexes, Acta Mathematica 201 (2008), 83-146.
- [FZ02] S. Fomin et A. Zelevinsky. *Cluster algebras I : Foundations*, Journal of the American Mathematical Society, Volume 15, Numéro 2 (2002), 497-529.
- [Hat91] A. Hatcher. *On Triangulations of Surfaces*, Topology and its Applications 40 (1991), 189-194.
- [Mas89] W.S. Massey. *Algebraic topology : an introduction*, 2^e éd., Springer (1989), 261 p.
- [Sch79] H. Schwerdtfeger, *Geometry of Complex Numbers*, Courier Dover Publications (1979), 215 p.
- [Tot02] G. Toth. *Stereographic Projection*, Glimpses of Algebra and Geometry, 2^e éd., Springer (2002).

GUILLAUME DOUVILLE, DÉPARTEMENT DE MATHÉMATIQUES, UNIVERSITÉ DE SHERBROOKE
 Courriel: `Guillaume.Douville@usherbrooke.ca`

FONCTIONS DE FRISES ET ALGÈBRES AMASSÉES

MYRIAM CHABOT

RÉSUMÉ. À partir d'un carquois fini et acyclique, il est possible de construire la répétition de ce carquois et de définir ce que sont les fonctions de frise. En étudiant quelques exemples, on dégagera quelques propriétés des fonctions de frise et on explicitera certains liens existant avec les algèbres amassées et les polynômes continuants signés.

1. Introduction

Introduites par Fomin et Zelevinsky au début du siècle, les algèbres amassées constituent une branche de la mathématique en plein essor à laquelle se rattachent plusieurs sujets comme la théorie des représentations, les systèmes intégrables et la thermodynamique [Ngu06]. À partir d'un **carquois** fini sans boucles ni 2-cycles et d'un ensemble de variables appelé **amas initial** et dont les variables sont appelées des **variables amassées**, il est possible d'engendrer de nouveaux carquois et de nouvelles variables amassées à l'aide du concept de **mutation**. En regroupant l'ensemble des variables ainsi formées, en nombre fini ou infini, nous obtenons les générateurs d'une algèbre appelée l'algèbre amassée. L'étude des algèbres amassées a permis d'aboutir à plusieurs résultats tels le phénomène Laurent [FZ02] et la classification des algèbres amassées de type fini [FZ03]. Plusieurs résultats restent à découvrir ou à démontrer, par exemple la conjecture de positivité.

Dans cet article, nous verrons comment il est possible de lier les idées d'algèbres amassées et de frises. Bien que l'élaboration de frises remonte aux grandes civilisations égyptiennes, elles ne sont étudiées comme sujet mathématique que depuis peu [RS08]. Ainsi, en partant d'un carquois, nous pouvons construire la **répétition** de ce carquois. Découle alors de cette construction la notion de **fonction de frise**, qui constitue notre sujet d'intérêt et qui permet d'explorer sous un autre angle les algèbres amassées.

Merci à Ibrahim Assem, Thomas Brüstle, Virginie Charette, Grégoire Dupont, Vasilisa Shramchenko et David Smith pour les séminaires sur les algèbres amassées. Merci à David Boulet-St-Jacques, Guillaume Douville et Philippe Dompierre pour les nombreuses discussions qui ont contribué à l'élaboration de cet article. Je remercie également l'ISM pour le financement d'une bourse de premier cycle.

L'objectif de cet article est de dresser la liste de quelques propriétés des fonctions de frises et de leurs liens avec les algèbres amassées. Pour ce faire, nous expliciterons tout d'abord ce que sont les fonctions de frise pour ensuite en étudier quelques-unes qui exposeront certains phénomènes récurrents. En effet, les exemples présentés serviront de point de départ pour dégager certaines propriétés gravitant autour des fonctions de frise, comme la périodicité et la nature de l'ensemble des valeurs générées par une fonction de frise. Puis, nous constaterons que certaines propriétés peuvent être interprétées comme conséquence de résultats sur les algèbres amassées. Finalement, une brève introduction aux polynômes continuants signés permettra d'étudier les fonctions de frise dans le cas particulier d'un carquois dont le graphe sous-jacent est de type \mathbb{A}_n .

2. Définition d'une fonction de frise

Afin d'introduire le concept de fonction de frise, présentons une définition intuitive des frises¹. Omniprésente dans ce monde, la frise est un motif se répétant sur une surface de longueur infinie, mais de largeur finie et constante. Étant périodique, à chaque frise est associée une longueur minimale qui la laisse invariante à la suite d'une translation dans le sens longitudinal [RS08]. Appliquons ce concept aux carquois.

Notation 2.1. *Considérons le graphe G . On notera \vec{G} le carquois linéairement orienté, c'est-à-dire G avec une orientation linéaire.*

Pour appliquer la notion de frise aux carquois, prenons l'exemple du carquois \vec{A}_n et répétons ce carquois un nombre infini de fois.

Nous avons ainsi obtenu une frise à partir du carquois \vec{A}_n . Toujours dans l'optique d'en arriver aux fonctions de frise, tentons de relier chacune de ces copies de carquois tout en conservant la périodicité. Il faut alors définir de nouvelles flèches entre les carquois obtenus. Ceci nous amène à définir la répétition \vec{A}_n^{rep} d'un carquois \vec{A}_n fini et acyclique. Rappelons que pour un carquois \vec{A}_n , V est l'ensemble des sommets, E est l'ensemble des flèches et s, t sont des fonctions $V \rightarrow V$, où s représente la source d'une flèche et t son but.

1. Pour une présentation plus rigoureuse des frises, voir [BCD11].

Définition 2.2.

Soit \mathcal{C} un carquois fini et acyclique. Sa **répétition** $\mathcal{C}^{\text{rép}}$ est le carquois défini par

Autrement dit, pour construire la répétition $\mathcal{C}^{\text{rép}}$ d'un carquois fini et acyclique \mathcal{C} , on répète le carquois \mathcal{C} en conservant tous les points et toutes les flèches qui le composent. De plus, dans la i -ième copie de \mathcal{C} , s'il y a une flèche α du sommet s au sommet t , alors il y aura une flèche $\alpha + i$ du sommet s de la i -ième copie au sommet t de la i -ième copie.

Remarque. La i -ième copie d'un carquois \mathcal{C} est celle située dans le sens croissant de la droite réelle, c'est-à-dire celle à droite de la $(i-1)$ -ième copie.

Afin d'illustrer cette définition, voici deux exemples de répétitions de carquois.

Exemples 2.3.

a) Soit \mathcal{C} le carquois $\begin{array}{ccc} & \bullet & \\ & \downarrow & \\ \bullet & & \bullet \end{array}$. Alors la répétition $\mathcal{C}^{\text{rép}}$ est :

b) Soit \mathcal{C} le carquois $\begin{array}{ccccc} & & \bullet & & \\ & \swarrow & & \searrow & \\ \bullet & & & & \bullet \end{array}$. Alors la répétition $\mathcal{C}^{\text{rép}}$ est :

c) Soit \mathcal{C} le carquois $\begin{array}{ccccc} & & \bullet & & \\ & \swarrow & & \searrow & \\ \bullet & & & & \bullet \end{array}$. Alors la répétition $\mathcal{C}^{\text{rép}}$ est :

Définition 2.4.

À chaque $i \in \mathbb{Z}$, on associe le sous-carquois plein formé de tous les sommets et flèches situés entre le sommet s_i et le sommet s_{i+1} inclusivement. Ce sous-carquois est appelé **maille**. Graphiquement, une maille peut être

Remarque. Il serait également possible de numéroté les sommets en suivant une **numérotation admissible selon les sources** à la seule différence que l'on aurait

$$f_{i,j} = \begin{cases} 1 & \text{si } i < j \\ 0 & \text{sinon} \end{cases} \quad \text{pour toute flèche } i \rightarrow j.$$

Tout carquois fini et acyclique possède au moins une numérotation admissible. En effet, un carquois fini et acyclique possède nécessairement au moins un puits. Une fois un puits étiqueté, le sous-carquois restant, également acyclique, possède aussi au moins un puits, et ainsi de suite. Les exemples qui suivent sont tous des carquois dont les sommets sont numérotés admissiblement selon les puits.

Exemples 2.9.

a) 

b) 

c) 

Lemme 2.10. Soit une fonction de frise f sur un carquois fini et acyclique Q . Alors f est uniquement déterminée par ses valeurs sur l'ensemble des puits de Q .

DÉMONSTRATION. Soit Q un carquois fini et acyclique dont les sommets sont numérotés admissiblement selon les puits. Supposons que f est connue pour tout $i \in P$. Montrons que f est uniquement déterminée pour tout $i \in V$ et pour tout $j \in V$. On le montrera seulement pour $i < j$; le cas $i > j$ est semblable. Le cas $i = j$ étant trivial, supposons que f est uniquement déterminée par les valeurs sur P , pour $i < j$ et montrons-le pour $i < j$. Commençons par $i = 1$. Selon la définition, on a que

$$f_{1,j} = \sum_{i=1}^{j-1} f_{1,i} f_{i,j}.$$

Puisque le sommet 1 est un puits de Q , en raison de la numérotation admissible, alors toutes les flèches entrantes en j proviendront nécessairement de la copie j du carquois Q . Alors

,

d'où

$$f_{1,j} = \sum_{i=1}^{j-1} f_{1,i} f_{i,j},$$

un produit qui est uniquement déterminé par les valeurs de l'ensemble P en vertu de l'hypothèse de récurrence sur j . Supposons alors que f est connue et uniquement déterminée par les valeurs de l'ensemble P pour tout $i < j$.

En raison de la numérotation admissible, une maille commençant en n est de la forme

$$\vdots$$

Selon la définition, on a que

Donc f_n est uniquement déterminée par les f_k , pour $k \leq n$.

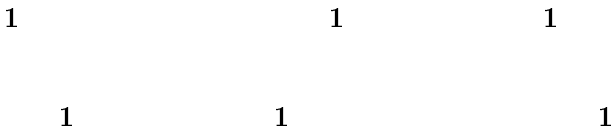
De cette manière, pour construire une fonction de frise, il suffit de donner des valeurs arbitraires aux éléments de l'ensemble \mathbb{Z} et de générer les autres au moyen de la relation de récurrence exposée à la définition 2.5.

3. Étude de certaines fonctions de frise

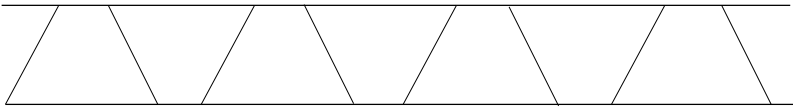
Voici 4 exemples qui seront par la suite étudiés afin d'énoncer certaines propriétés des fonctions de frise.

Exemples 3.1.

a) Soit \mathcal{C} le carquois \mathcal{C}_1 . Une fonction de frise unitaire sur \mathcal{C}_1 est



On remarque que l'image de la fonction de frise unitaire de ce carquois ne contient que des **entiers strictement positifs**, c'est-à-dire \mathbb{N}^* . En outre, il semble exister une certaine **périodicité** de forme trapézoïdale qui se répète successivement selon une réflexion à partir de l'axe horizontal de la frise, comme l'indique l'image suivante :



Deux thèmes reviennent souvent d'exemple en exemple, soient la périodicité et l'image de la fonction de frise. Tout d'abord, on remarque que pour les exemples , et , il n'y a que des valeurs entières et positives alors que l'image de la fonction de frise du deuxième exemple est plutôt comprise dans l'ensemble des rationnels. Remarquons que cet exemple est le seul pour lequel la fonction de frise n'est pas unitaire. Y aurait-il un lien ? Le fait d'utiliser une fonction de frise unitaire nous assure-t-il que l'image ne soit constituée que d'entiers positifs ?

Ensuite, en ce qui a trait à la périodicité, seul le dernier exemple ne possède pas cette propriété. En effet, on peut observer que l'image est en fait composée de certains nombres de la suite de Fibonacci, qui, on se le rappelle, est définie par la relation de récurrence , avec . Ainsi, on a

Donc la fonction de frise du carquois de Kronecker correspond à la suite , c'est-à-dire aux nombres de Fibonacci de rang pair. Or, les nombres de la suite de Fibonacci tendent vers l'infini, donc il ne peut pas y avoir de périodicité. Ainsi, on remarque une périodicité pour les trois premiers carquois, donc l'ensemble image est fini. Par contre, l'image de la fonction de frise est un ensemble infini en ce qui a trait au carquois , en raison de l'absence de périodicité. Puisque celui-ci est le seul à ne pas être construit à partir d'un carquois dont le graphe sous-jacent est de Dynkin, y aurait-il un lien entre la périodicité et les carquois issus de graphes de type Dynkin ?

De cette manière, il serait intéressant de catégoriser les situations dans lesquelles on a une fonction de frise d'entiers positifs et/ou périodique. Les prochaines sections élaboreront certains résultats qui permettront d'y parvenir.

4. Lien entre les fonctions de frise et les variables amassées

Cette section permettra d'expliquer certaines propriétés des fonctions de frise, et ce, à l'aide du lien qui existe avec les variables amassées. Pour ce faire, on rappelle quelques notions de base sur les algèbres amassées, en suivant les définitions de [Bou11].

Pour une graine , la **mutation** en une variable transforme l'amas en un nouvel amas , où

$$— \quad \quad \quad .$$

De plus, la **mutation d'un carquois** en un sommet transforme en un nouveau carquois obtenu au moyen des étapes suivantes :

- (1) Pour tout chemin de longueur 2 de type , on ajoute une flèche du sommet vers le sommet .
- (2) On renverse le sens de toutes les flèches incidentes à .
- (3) On élimine tous les 2-cycles.

Définition 4.1.

On appelle **réflexion** une mutation de carquois appliquée à une source ou un puits.

Ainsi, réfléchir un carquois en une source ou un puits revient simplement à inverser les flèches incidentes à .

Considérons l'amas initial et la graine initiale . Puisque le graphe sous-jacent à ce carquois est , alors il y aura 5 variables amassées, car pour un graphe , il y a variables amassées [Ngu06]. On remarque qu'en appliquant uniquement des réflexions à l'unique source du carquois , on obtient toutes les variables amassées :

$$\begin{aligned} & , \text{ où } \text{-----} \\ & , \text{ où } \text{-----} \\ & , \text{ où } \text{-----} \end{aligned}$$

L'algèbre amassée associée au carquois est alors

$$\text{-----} \text{-----} \text{-----} .$$

En observant la répétition , on remarque qu'elle est obtenue par une suite de réflexions appliquées à la seule source du carquois .

Par exemple, en considérant le carquois de départ , on obtient le carquois à la suite d'une mutation appliquée en la source . Ainsi, en posant pour tout , on obtient la fonction de frise suivante :

Toutes les variables amassées qui se trouvaient dans sont dans ce carquois. De cette manière, en plus de calculer des valeurs numériques pour les sommets, les fonctions de frises nous permettent de calculer des variables amassées obtenues par les mutations d'un carquois fini et acyclique, comme l'indique le théorème suivant.

Théorème 4.2. Soient un carquois fini et acyclique et une fonction de frise. On pose , pour tout . Alors,

- a) Pour tout , est une variable amassée. Une telle frise est appelée une **frise de variables**.

- b) Si Γ est un carquois dont le graphe sous-jacent est de type Dynkin, alors toutes les variables amassées se retrouvent dans la frise de variables de

DÉMONSTRATION. Voir, par exemple, [ARS10].

En outre, à l'exemple 3.1, nous avons obtenu que les valeurs de la fonction de frise des carquois Γ_1 , Γ_2 et Γ_3 n'étaient formées que d'entiers positifs. Par ailleurs, il ne s'agissait que de frises unitaires. Voici un énoncé permettant de relier ces deux concepts.

Corollaire 4.3. *Soit Γ un carquois fini et acyclique. Une fonction de frise unitaire de Γ ne contient que des entiers.*

DÉMONSTRATION. En vertu d'un théorème de Fomin-Zelevinsky [FZ02], pour une graine quelconque s d'une algèbre amassée \mathcal{A} , toute variable amassée s'écrit comme un polynôme de Laurent en les variables x_i . Donc x_i est une fraction de polynômes. En considérant la frise de variables associée à s , on sait qu'elle ne contiendra que des variables amassées, qui peuvent donc s'écrire sous la forme :

$$\frac{p}{q}, \text{ où } p, q \text{ sont des polynômes en les } x_i \text{ pour tout } i.$$

Pour obtenir la frise unitaire associée à s , il suffit d'évaluer x_i en 1, pour tout i compris entre 1 et n . Ainsi, le dénominateur de $\frac{p}{q}$ sera toujours 1, et le numérateur sera un entier. Donc $\frac{p}{q}$ est un entier.

Toutefois, on remarque que le corollaire 4.3 ne nous assure pas que les valeurs d'une fonction de frise unitaire seront des entiers positifs, alors que c'était le cas dans les carquois Γ_1 , Γ_2 et Γ_3 de l'exemple 3. Ainsi, nous pouvons faire un autre lien avec les variables amassées, plus précisément, la conjecture de positivité. On rappelle que la conjecture de positivité dit que pour une graine quelconque s d'une algèbre amassée \mathcal{A} , toute variable amassée s'écrit sous la forme

$$\frac{p}{q}, \text{ avec } p, q \text{ des polynômes en les } x_i \text{ et } q \text{ sans terme constant.}$$

Ainsi, cette conjecture nous assurerait que les valeurs d'une fonction de frise unitaire seraient comprises dans les entiers strictement positifs. En 2009, Musiker, Schiffler et Williams ont démontré cette conjecture pour toute algèbre amassée provenant d'une surface marquée [MSW09], ce qui inclut les cas Γ_1 , Γ_2 et Γ_3 [Bou11]. Nous pouvons alors élargir le corollaire 4.3 aux cas dans lesquels la conjecture de positivité a été démontrée.

Corollaire 4.4. *Soit Γ un carquois fini et acyclique provenant d'une surface marquée. Une fonction de frise unitaire de Γ ne contient que des entiers positifs.*

DÉMONSTRATION. Cela suit du corollaire 4.3 et de la conjecture de positivité [MSW09].

De plus, dans le cas particulier des carquois dont le graphe sous-jacent est de type \tilde{A} , il est possible de généraliser ce dernier corollaire. En effet, pour une fonction de frise f telle que

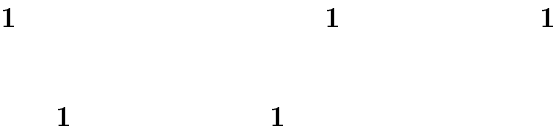
$$f(x) \in \mathbb{Z}$$

alors les valeurs de la fonction de frise f sont des entiers strictement positifs [CC73].

5. Récurrences linéaires

Cette section montre comment il est possible de prédire récursivement les valeurs d’une fonction de frise. Pour ce faire, à des fins de simplification, on n’utilisera dans nos exemples que des frises unitaires.

Pour introduire ce concept, reprenons un exemple déjà présenté dans la section 3. On avait le carquois Q , que l’on réécrira sous la forme Q' . La fonction de frise unitaire était :



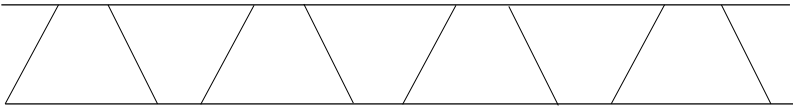
Ainsi, on remarque que pour chaque sommet i , il y a une suite de nombres $f(i)$ associés à i dans la fonction de frise. Effectivement, on a la suite $f(i)$ pour les valeurs de i et $f(i)$ pour les valeurs de i . Ceci nous mène à appliquer la notion de récurrence linéaire aux fonctions de frise.

Définition 5.1.

Soit $(a_n)_{n \in \mathbb{N}}$ une suite d’éléments d’un corps K . Alors la suite satisfait à une **récurrence linéaire** d’ordre d , où $d \geq 1$, s’il existe des entiers c_1, \dots, c_d tels que

$$a_n = c_1 a_{n-1} + \dots + c_d a_{n-d}$$

On avait remarqué à la section 3 que la périodicité était sous forme trapézoïdale, comme l’indique l’image ci-dessous :



où, dans l’exemple 3.1 a), un trapèze contenait les nombres suivants :

Ainsi, dans le cas \mathcal{C}_n , on peut voir que la récurrence linéaire se définit comme suit :

En outre, on peut généraliser cette récurrence linéaire à tout carquois \mathcal{C} , où \mathcal{C} est un carquois fini. En effet, pour la fonction de frise $f_{\mathcal{C}}$ et un sommet i de \mathcal{C} , la récurrence linéaire est :

c'est-à-dire

Voici d'autres exemples de fonctions de frise pour lesquelles la récurrence linéaire sera étudiée. Un d'entre eux a déjà été présenté dans la section 3.

Exemples 5.2.

- a) Soit le carquois de Kronecker \mathcal{C}_n . Sa fonction de frise unitaire est alors

On a vu dans l'exemple 3.1 que les nombres obtenus dans la fonction de frise unitaire de \mathcal{C}_n étaient les nombres de Fibonacci de rang pair. Or, on peut facilement vérifier que les nombres de Fibonacci de rang pair sont donnés par la relation de récurrence $F_{2n} = F_{2n-2} + F_{2n-4}$, où $F_0 = 0$ et $F_2 = 1$. En effet,

De cette manière, on obtient que

On en déduit que

Précédemment, nous avons déjà remarqué que pour les carquois dont le graphe sous-jacent était de type Γ , la périodicité était de forme trapézoïdale. Conway et Coxeter ont d'ailleurs montré comment s'exprimait cette périodicité.

Théorème 5.4. [CC73] *Soit Γ un carquois fini et acyclique. Soit f une fonction de frise. Alors la suite de nombres $(f_n)_{n \geq 0}$ vérifie une récurrence linéaire si et seulement si Γ est de type Dynkin ou Euclidien.*

Le dernier théorème présenté dans cette section permettra d'affirmer s'il y a présence ou non d'une récurrence linéaire seulement en observant le type de carquois. En 2010, les travaux de Assem, Reutenauer et Smith et ceux de Keller et Scherotzke sur la récurrence linéaire ont permis d'arriver à certains résultats à ce sujet.

Théorème 5.5. ([ARS10], [KS10]) *Soient Γ un carquois fini et acyclique, v un sommet de Γ et f une fonction de frise. Alors la suite de nombres $(f_n)_{n \geq 0}$ vérifie une récurrence linéaire si et seulement si Γ est de type Dynkin ou Euclidien.*

La nécessité a été prouvée dans [ARS10] en utilisant les concepts de fonctions additives et sous-additives. D'ailleurs, il a également été démontré que pour une fonction de frise f , celle-ci était bornée si Γ était de type Dynkin et non-bornée dans le cas où Γ était de type Euclidien ([ARS10], Theorem 1). En ce qui a trait à la suffisance, le cas Dynkin est trivial en raison de la proposition 5.3. Puis, Assem, Reutenauer et Smith ont prouvé les cas Γ et Γ de manière combinatoire. Il s'agissait d'une conjecture dans les autres cas. Par contre, en ayant recours à une autre méthode, Keller et Scherotzke ont réussi à le démontrer pour tous les cas de type Euclidien [KS10].

On déduit du théorème précédent qu'il n'y aura pas de récurrence linéaire pour l'exemple 5.2 c), car le graphe sous-jacent n'est ni de type Dynkin ni de type Euclidien.

6. Polynômes continuants signés

Cette section présentera un lien qui existe entre les fonctions de frise et les polynômes continuants signés.

Définition 6.1.

Soit $(x_i)_{i \in \mathbb{Z}}$ une famille infinie de variables. Le **polynôme continuant signé**, ou **polynôme de Tchebychev généralisé**, est donné par la relation de récurrence

$$P_0 = 1,$$

avec $P_1 = x_1$ et où n est le nombre de variables.

Théorème 6.3. ([FL10], Cor. 3.14) *Soit \mathcal{F} une fonction de frise. Alors,*

Ces résultats ont été appliqués dans les travaux de Dupont en lien avec les tubes stables de rang n et la fonction de Caldero-Chapoton [Dup09].

Références

- [ARS10] Assem, I., Reutenauer, C. et Smith D. *Friezes*, Advances in Mathematics 225 (2010) 3134-3165.
- [Bou11] Boulet-St-Jacques, D. *Les algèbres amassées : Définitions de base et résultats*, (2011) soumis pour publication.
- [BCD11] Boulet-St-Jacques, D., Chabot, M. et Douville G. *Groupes de frises des répétitions des carquois de type Dynkin*, (2011) soumis pour publication.
- [CC73] Conway, J.H. et Coxeter, H.S.M. *Triangulated polygons and frieze patterns*, The Mathematical Gazette 57, no. 400, (1973) 87-94.
- [Dup09] Dupont, G. *Cluster Multiplication in Regular Components via Generalized Chebyshev Polynomials*, arXiv :0801.3964v2 [math.RT] (2009) 1-20.
- [FL10] Fraser Martineau, J.-S. et Lavertu, D. *Frises et triangulations de polygones*, Revue CaMUS 1, Université de Sherbrooke, Département de mathématiques, (2010) 39-59.
- [FZ02] Fomin, S. et Zelevinsky, A. *Cluster algebras I : Foundations.*, Journal of the American Mathematical Society, Volume 15, Numéro 2, (2002) 497-529.
- [FZ03] Fomin, S. et Zelevinsky, A. *Cluster algebras II : Finite type classification*, Invent. Math. 154 (2003), no. 1, 63-121.
- [KS10] Keller, B. et Scherotzke, S. *Linear recurrence relations for cluster variables of affine quivers*, arXiv :1004.0613v2 [math.RT], (2010) 1-20.
- [MSW09] Musiker, G., Schiffler, R. et Williams, L. *Positivity For Cluster Algebras From Surfaces*, arXiv :0906.0748v1 [math.CO], (2009) 1-67.
- [Ngu06] Nguefack, B. *Introduction aux Algèbres amassées : Définitions et exemples*, http://www.usherbrooke.ca/mathematiques/fileadmin/sites/mathematiques/documents/Recherche/Publications/Rapports_de_recherche/rr41.pdf, (2006) 1-33.
- [RS08] Rousseau, C. et Saint-Aubin, Y. *Mathématiques et technologies*, Springer Undergraduate Texts in Mathematics and Technology (2008) chapitre 2, 45-84.

MYRIAM CHABOT, DÉPARTEMENT DE MATHÉMATIQUES, UNIVERSITÉ DE SHERBROOKE
Courriel: Myriam.Chabot@usherbrooke.ca

ANALYSE ET COMPARAISON DE PROCÉDURES D’OPTIMISATION EN TOMODENSITOMÉTRIE

MAXIME TOUSSAINT ET CHRISTIAN THIBAudeau

RÉSUMÉ. La est une modalité d’imagerie médicale utilisée pour représenter la structure interne d’un être vivant. La reconstruction de cette image doit être rapide et stable. En outre, le résultat doit représenter le plus fidèlement que possible la réalité. Ce document propose de comparer les performances de quelques algorithmes itératifs en termes de qualité d’image reconstruite et de durée de calcul. Parmi les algorithmes testés, certains ont été développés spécialement pour le problème de tomodensitométrie alors que d’autres sont des solveurs généraux. De plus, chacun des algorithmes sera testé avec une variante incluant un terme de régularisation. Ce document propose de tester quelques variantes de gradient conjugué non-linéaire afin d’observer leur performance en tomodensitométrie.

1. Introduction

La tomodensitométrie est une procédure qui consiste à exploiter le taux d’absorption de rayons-x des tissus pour représenter la structure interne d’un être vivant. Ce procédé se divise en trois étapes, soit l’acquisition, le traitement des données et la reconstruction. Ce document se concentrera sur cette dernière étape dans le cas particulier d’une géométrie d’acquisition en seulement 2 dimensions.

Dans la littérature, deux familles de méthodes sont souvent reliées à la tomodensitométrie. Les méthodes analytiques qui reposent sur l’inverse de la transformée de Radon et les itératives qui optimisent des modèles correspondant au problème. Ces deux procédés sont possibles grâce à un théorème démontré par M. Johann Radon en 1917 [RP86].

Théorème de projection de Radon : Si la totalité des projections d’un objet est disponible et que ces dernières sont acquises selon des droites concourantes, il est alors possible de reconstruire la fonction réelle à deux variables (position spatiale en x et en y) représentant cet objet.

Cet article provient d’un travail qui a été financé par une bourse offerte par la Faculté de médecine et des sciences de la santé de l’Université de Sherbrooke (FMSS), les Instituts de recherche en santé du Canada (IRSC) et le Conseil de recherches en sciences naturelles et en génie du Canada (CRSNG). Les auteurs tiennent à remercier M. Roger Lecomte pour leur avoir permis de travailler sur ce projet et M. Jean-pierre Dussault pour les commentaires et les discussions qui ont motivé l’ajout des SG. Leur critique ont permis d’améliorer la rédaction de ce rapport technique.

En pratique, il est impossible d'obtenir toutes les projections. Toutefois, un sous-échantillon judicieusement choisi permet d'acquérir suffisamment de données pour conserver l'idée du théorème. De plus, les limitations des ressources informatiques ne permettent pas de représenter l'objet dans sa forme continue. La discrétisation en voxels permet de pallier à cette lacune. Pour plus d'informations concernant l'acquisition et le traitement des données, le lecteur est invité à consulter [KS88] et [H80].

Deux caractéristiques importantes en tomodesitométrie différencient les algorithmes analytiques de ceux dits itératifs : la durée de reconstruction et la flexibilité de la méthode. Pour se représenter la différence, il faut comprendre leur fonctionnement. La reconstruction par calcul analytique consiste à appliquer la fonction inverse sur les résultats obtenus. L'idée de base veut que la méthode d'acquisition des données puisse être représentée par une fonction qui, en utilisant le théorème de projection de Radon, admet une inverse. Le principe clé des algorithmes itératifs est de représenter la problématique sous forme d'un problème d'optimisation et de chercher, à chaque étape, une solution qui se rapproche de la vérité. Il s'agit ensuite d'arrêter lorsque cette solution est jugée satisfaisante.

Pour comparer la vitesse de reconstruction des familles d'algorithmes, il faut introduire l'élément le plus coûteux à manipuler, soit la matrice système. Cette matrice permet de relier les données acquises à la version discrète de l'objet. Elle devient rapidement imposante puisque sa forme explicite est composée de $N \times P$ éléments (où N représente le nombre de projections, P le nombre de détecteurs et V le nombre de voxels). Ainsi, la vitesse de reconstruction des méthodes dépendra largement de la taille de cette matrice et du nombre de fois qu'elle doit être utilisée. La méthode analytique demande l'application d'une formule, soit l'inverse de la fonction de Radon, ce qui consiste à n'utiliser qu'une seule fois la matrice système pour approximer l'objet. Une méthode itérative demande l'application de cette matrice au minimum une fois par itération. Donc, une méthode itérative demande beaucoup plus de calcul qu'une méthode analytique.

Toutefois, les algorithmes itératifs sont connus pour être plus flexibles que les algorithmes analytiques. D'une part, il est possible d'inclure dans ces premiers la mathématique liée à la physique du processus d'acquisition. Par exemple, la nature poissonnienne du comptage de photons. D'un autre côté, alors que les algorithmes analytiques doivent travailler avec des données spatialement équidistribuées (et souvent uniquement en géométrie parallèle), les méthodes itératives ne souffrent d'aucunes limitations de ce type. En modélisant correctement la géométrie du système d'acquisition, par le biais de la matrice système, toutes les configurations possibles de scanners peuvent être utilisées pour produire des images sans ré-interpolation des données. Les algorithmes itératifs sont principalement composés de deux éléments. D'abord, il faut un modèle qui se comporte bien, c'est-à-dire continu et dérivable, et dont l'image recherchée correspond à la solution optimale. Puis, il faut une méthode d'optimisation qui converge vers la solution désirée. Les détails seront expliqués dans la section suivante.

Le but de ce document est de comparer les performances de différentes méthodes itératives. Afin d'offrir une vue d'ensemble sur les méthodes disponibles en tomodesitométrie, des algorithmes reconnus dans le domaine ont été employées

ainsi que des variantes de deux solveurs généraux qui semblent prometteuses. Ainsi, la liste est composée de méthodes développées uniquement pour résoudre des problèmes de tomographie et de solveurs généraux. Il est à noter que dans ce travail, ces deux groupes sont respectivement nommés APT¹ et SG². De plus, chacun des algorithmes sera testé avec et sans un terme de régularisation, ce qui permettra de comparer leurs performances. Pour finir, les expériences incluent un APT qui se sert de sous-groupes ordonnés, une reconnue dans la littérature pour sa rapidité de convergence.

Dans ce court rapport, les algorithmes testés dans les expériences seront d'abord présentés. Ensuite, les mires utilisées dans les simulations seront décrites. Suivra la description de la méthodologie appliquée pour effectuer les tests et les comparaisons. Les résultats, appuyés par des graphiques, viendront illustrer les performances des algorithmes. Finalement, le document va clore sur l'évaluation des performances.

Cet article est une version condensé du rapport de recherche [TTL11]. La version complète inclut la description et la motivation des algorithmes, le code des mires, les majorations, où une fonction majore lorsque pour tout , , utilisées et quelques ouvertures dans ce domaine.

2. Algorithmes utilisés

Les algorithmes utilisés dans ce travail sont résumés dans le tableau ci-dessous et la notation utilisée dans cet article est présentée à la section 6.1.

Notez que le code de L-BFGS-B utilisé se retrouve dans le lien suivant :

http://blake.bcm.edu/eman2/doxygen_html/lbfgsb_8cpp_source.html, que la notation GC fait référence au gradient conjugué non-linéaire et que P&R et H&Z font respectivement référence à Polak & Ribière et Hager & Zhang.

Notation	Nom usuel	Référence
Conv	Convexe	Voir [LF95]
Gradient	Gradient avec longueur admissible d'Armijo	
trml	Espérance-maximisation	
Conv_os	Convexe avec 10 OS	Voir [KB05]
GC	GC avec le Bêta de P&R sur	Voir [HZ05]
GC_HZ	GC avec le Bêta de H&Z sur	
GC_HZ*	GC avec le Bêta de H&Z sur	
GC*	GC avec le Bêta de P&R sur	
L-BFGS-B	L-BFGS-B où la fonction objectif est	Code
L-BFGS-B*	L-BFGS-B où la fonction objectif est	de EMAN2

Fonction objectif :

—

1. APT : Algorithmes Produits pour la Tomodensitométrie

2. SG : Solveurs généraux

où ℓ est la log-vraisemblance de la solution courante selon le sinogramme, voir [LC84], et γ , présenté dans [SB93], est une approximation quadratique de ℓ .

Fonction de pénalisation :

Le but de cette fonction est de favoriser les solutions formées de zone uniforme. Le gamma contrôle la force globale tandis que l'épsilon contrôle le point critique où la fonction passe de quadratique à linéaire.

Formule :

$$F(x) = \begin{cases} \gamma(x) & \text{si } \gamma(x) \leq \epsilon \\ \gamma(x) + \frac{1}{2\epsilon}(\gamma(x) - \epsilon)^2 & \text{sinon} \end{cases}$$

Dans ce travail, l'épsilon a été choisi de sorte à ce que $\gamma(x) \leq \epsilon$. Soit

$$x^* = \arg \min_x F(x)$$

Les quatre premiers algorithmes sont modifiés tel présenté en [LF95] lors de l'ajout d'un terme de pénalisation. Pour les autres, il suffit d'ajouter $F(x)$ à la fonction objectif.

Fonction barrière :

L'algorithme du gradient conjugué non linéaire ne tient pas compte, dans sa définition, des contraintes. Une des méthodes pour palier à ce problème est d'ajouter une fonction barrière à la fonction objectif. Pour plus d'information, consultez [D95]. La fonction utilisée est $B(x) = \frac{1}{\beta} \log \frac{1}{\beta(x)}$. L'algorithme utilisé pour calculer la longueur admissible est explicité dans [CMI09]. Cet algorithme demande une majoration de la courbure et un β qui représente la force de la barrière et qui est diminué lorsque la solution du problème courant est atteinte.

Majoration utilisée : Les étapes sont explicitées en [TTLD11]. dir est la direction courante de l'algorithme.

$$\beta = \beta_0 \exp(-\alpha \sum_{i=1}^n \langle dir, \nabla \beta(x_i) \rangle)$$

Ordered Subset (OS) : Le principe est de séparer les projections en quelques groupes et de mettre à jour, à chaque itération, la solution avec chacun des sous-groupes. L'algorithme est très rapide puisque le problème est divisé en morceaux, mais il tourne autour de la solution optimale, puisqu'il converge vers les solutions individuelles des sous-problèmes.

3. Description des images tests utilisées

Afin de comparer les performances des algorithmes dans différentes situations, les données vont être simulées à partir de 4 objets synthétiques. Les mires ont été produites par CTSim³ et les sinogrammes ont été créés par simulation à l'aide du programme SpectroSin développé par M. Christian Thibaudau. La simulation a été produite dans quatre conditions différentes. Soit 100, 1 000 et 10 000 photons incidents par projection en ajoutant un bruit de nature Poissonienne⁴ et une version sans bruit.

Pour faciliter et uniformiser l'évaluation des critères de comparaison, les objets ont été construits avec quelques propriétés. Tout d'abord, les objets peuvent tous être insérés dans un plan de 20 par 20 cm. Le calcul de certains des critères de comparaison utilisés dans ce projet exige une zone uniforme, notée U , et une mince ligne, notée L , avec un coefficient d'atténuation élevé. Voir la section 6.2 pour observer leur position dans les mires. Les valeurs des coefficients d'atténuation des mires ont été ajustées afin que la zone qui inclut U ait la même amplitude dans les quatre mires. Les mires sont affichées à la sections 6.4.

Shepp-Logan : La mire du Shepp-logan, dont la définition peut être trouvée dans [KS88], représente une coupe du crâne humain. À l'intérieur de celui-ci, se trouvent des zones d'atténuation différentes correspondant à des formes typiques de ce que l'algorithme doit être capable de reproduire.

Shepp-Logan facile : La structure de cette mire est identique à la précédente. Cependant, les valeurs des coefficients d'atténuation ont été modifiées afin de faciliter l'observation des contrastes. Ils consistent à augmenter de 25 ou 50 fois la différence entre les coefficients de la structure de base et ceux des autres zones.

H.G.D. : L'image a été inspirée de la mire utilisée dans [HGD11] afin de comparer les performances de certains algorithmes de reconstruction en tomographie. Elle est composée de plusieurs disques qui varient en taille et en atténuation. Cette mire permet d'évaluer les limites des algorithmes. Elle permet d'évaluer la limite d'amplitude et de surface qu'une zone uniforme doit avoir pour être visible dans la solution.

Catphan CTP401 : Les mires Catphan sont utilisées pour tester et ajuster la qualité des reconstructions. Habituellement, elles servent à tester l'équipement. Après une lecture d'un de leurs catalogues⁵, il a été décidé d'inclure dans les tests le module CTP401 afin d'obtenir un aperçu de la performance des algorithmes sur une structure utilisée dans le domaine. La mire utilisée est une approximation du module CTP401.

4. Analyses accomplies

L'objectif principal de ce document est de comparer l'efficacité des algorithmes. Or, il existe plus d'une technique pour comparer leurs performances. L'approche choisie dans ce projet est divisée en deux parties. Pour commencer,

3. Programme open source de simulation de tomographie, <http://ctsim.org/>

4. Le rapport de signal-sur-bruit est donné par

5. Il est accessible à la page web www.phantomlab.com/documents.php

elle propose d'observer le comportement des méthodes au fil des itérations. Ensuite, elle confronte les résultats des algorithmes afin d'en tirer un classement selon différents critères.

Cependant, cette dernière approche présente quelques difficultés. Tout d'abord, le comportement et les solutions obtenues par les méthodes sont potentiellement sensibles à la configuration de l'acquisition, à la solution initiale, à l'objet analysé et à la qualité des données obtenues. Les analyses de ce projet se limitent à une seule configuration⁶. Tous les tests sont initialisés avec une image dont toutes les valeurs d'atténuation sont égales à 1. La section 3 introduit les 16 problèmes que doivent résoudre les méthodes, soit 4 mires avec chacun 4 niveaux de bruit. Cet éventail d'expériences sera utilisé pour tenter de présenter des données indépendantes des mires et/ou de la qualité des acquisitions. D'autres obstacles se présentent lorsque les solutions des méthodes sont comparées. Il faut choisir un critère d'arrêt pour les algorithmes et définir une méthode pour évaluer le meilleur résultat. Notez que dans ce travail, la valeur du gamma est de 0.5 pour tous les algorithmes sauf lorsque les OS sont utilisés, où il vaut 0.1. Le gamma a été choisi après des tests effectués sur les APT. La modification apportée pour les OS est motivée par un désir de garder la même force de pénalisation pour tous les algorithmes et le fait que l'utilisation de 10 sous-groupes divise le problème de départ en 10 sous-problèmes. La force du epsilon est fixée à 0.01 et le γ correspondra à la plus petite différence d'atténuation non-nulle. Soit $\gamma = 0.01$, $\epsilon = 0.01$, et $\gamma = 0.1$, respectivement pour Shepp-Logan facile, Shepp-Logan, H.G.D et CTP401.

4.1. Observer l'évolution des solutions au fil des itérations

L'objectif derrière l'observation du comportement des méthodes est de fournir suffisamment d'information afin de pouvoir les classer. En vue d'accomplir cette tâche, cet article propose d'observer trois variables : le MSE⁷ entre la solution à une itération donnée et la mire pixelisée, la LMH⁸ de la ligne B et la variance de la zone A⁹. Les valeurs affichées sont les moyennes des valeurs sur les 16 tests. Les variables ont été enregistrées à toutes les 10 itérations, et ce jusqu'à la 100^{ème} itération. C'est en observant que les APT produisaient des images convenables en 80 itérations et le désir de ne pas faire trop d'itérations qui a motivé le choix de cette borne. Notez que dans ce document, une itération est définie comme une mise à jour de la solution courante. Donc, une application de la méthode convexe OS 10 est comptée pour 10 itérations. Afin que les résultats soient semblables entre les algorithmes, ils sont tous compilés aux 10 itérations. Les méthodes vont être séparées en deux graphiques selon qu'ils ont un terme de régularisation ou non, et ce, pour chaque variable. Ils seront respectivement nommés algorithmes pénalisés et algorithmes standards. L'ajout de $_P$ à un nom d'algorithme indique qu'il fait partie des algorithmes pénalisés. Le but est d'éviter

6. Voir section 6.3.

7. *Mean square error* ou erreur quadratique moyenne

8. La largeur à mi-hauteur est évaluée sur une ligne verticale de 18 voxels. Afin de diminuer l'effet du bruit sur l'évaluation du LMH, cette ligne est la moyenne de 22 lignes de même longueur et dont chacune intersecte perpendiculairement la ligne B. L'intersection divise les 22 lignes en deux parties égales.

9. Voir la section 6.2

de surcharger les graphiques et de permettre une comparaison rapide entre ces deux types d'algorithmes. Les graphiques sont tous d'une forme semblable à la figure 4a. Chaque algorithme est représenté par une ligne brisée d'une couleur indiquée dans la légende. Cette courbe représente la moyenne de l'évolution de la variable en question sur les 16 tests selon les itérations.

4.2. Comparer les solutions obtenues avec un critère d'arrêt

Pour être en mesure de comparer les résultats produits par chacune des méthodes, il faut déterminer un critère d'arrêt. Habituellement, l'objectif principal en tomodensitométrie est de reconstruire une image qui a une résolution satisfaisante. Une approche reconnue pour évaluer la résolution est de mesurer la LMH. De plus, des analyses préliminaires montrent que les algorithmes itératifs commencent par produire la forme générale de la mire¹⁰. Donc pour cette batterie de tests, les algorithmes itèrent jusqu'à ce que la LMH de la ligne B soit suffisamment petite. Toutefois, certains algorithmes convergent trop lentement vers une LMH acceptable ou même qu'ils commencent à croître au fil des itérations. Ainsi, les algorithmes ont été fixés à une limite de 200 itérations. Ceci correspond à fois plus que ce qui est nécessaire pour les algorithmes APT avant d'obtenir une solution très proche, à vue d'œil, de leur mire respective. De plus, des expériences préliminaires indiquent que les algorithmes avec un terme de régularisation ont des valeurs de LMH plus élevées. Cette différence est induite par l'effet de lissage du terme de pénalisation. Suite à d'autres tests, les choix de LMH sont fixés à 2 voxels et à 5 voxels pour les algorithmes standard et les *algorithmes_P* respectivement. Le critère d'arrêt étant choisi, il ne reste plus qu'à choisir les critères de comparaison. L'une des caractéristiques les plus importantes lorsque vient le temps de comparer des méthodes de reconstruction est la durée de calcul. Afin que les conclusions soient les plus générales possibles, les données affichées correspondront aux moyennes des 16 cas disponibles. Ensuite, le MSE et la variance dans la zone A seront calculés pour permettre d'évaluer qualitativement les solutions obtenues. Des diagrammes à bandes, tel que présenté en figure 11a, seront utilisés pour faciliter les comparaisons. La hauteur d'une bande représente la moyenne de la variable pour les tests présentés par cette dernière. Dans la figure 11a, la bande à l'extrémité gauche représente la moyenne des MSE pour les 4 mires lorsque reconstruites en utilisant l'algorithme convexe avec 10 OS, et ce, pour des données non bruitées. Les bandes sont triées en ordre croissant de bruit. Il s'agit des mêmes sous-groupes pour les graphiques sur la variance. Dans le graphique qui représente la durée de reconstruction des algorithmes, la bande à gauche correspond à la version standard de l'algorithme.

5. Résultats obtenus

Avant d'évaluer la capacité des algorithmes à produire une solution exacte et rapide, il faut observer comment ils se comportent généralement. Préalablement à la lecture des résultats, il faut spécifier certains détails. Les figures nommées

10. Ce constat n'est pas toujours vrai, et en conséquence, il a fallu demander un minimum d'itérations, 15 dans cette expérience, avant l'arrêt de l'algorithme.

dans cette section se retrouvent toutes dans la section 6.6. Les notations "*FObj1*" ou "*" font référence à la fonction objectif

5.1. Comportement des algorithmes sur 80 itérations MSE sur 80 itérations

Les figures 4a et 5a montrent que les solutions courantes des variantes du *GC* commencent loin de la solution de départ et peinent à rattraper les autres méthodes. Cette observation n'est pas sans fondement. Ces algorithmes sont les seuls qui doivent inclure un terme de barrière. Or, lors des tests, il n'y avait pas suffisamment d'information pour gérer la force de la barrière. Donc, il est possible qu'il existe un meilleur choix de λ qui améliorerait leur performance. De plus, il est possible de constater que les performances sont pires lorsqu'un terme de régularisation est ajouté. Cette constatation peut être expliquée par le choix de majoration qui n'est pas optimale. Puisqu'il est difficile de voir l'évolution des autres algorithmes dans ces figures, une figure correspondante, avec une échelle logarithmique a été jointe. La figure 4b montre une augmentation du MSE des algorithmes standards ce qui indique qu'ils s'éloignent des mires¹¹. Ce résultat est attendu puisque la solution optimale d'un sinogramme bruité est une image bruitée. L'ajout d'un terme de régularisation permet de corriger cette divergence (Figure 5a). Remarquez que la courbe de *convex* n'est pas visible dans ces figures. Cela est dû au fait qu'elle est trop près de la courbe de *convexos10*. Ainsi le fait d'utiliser environ 10 fois moins d'information par itération a permis d'obtenir des résultats quasi identiques à la méthode utilisant la totalité des données. On note aussi dans la figure 4b que les variantes de *L-BFGS-B* se comportent correctement au début, mais divergent de la solution voulue dès la 10^{ème} itération. Cette procédure est reconnue pour sa grande vitesse de convergence, ce qui n'est pas nécessairement voulu lorsque les données sont bruitées. La figure 5a montre que l'ajout d'un terme de régularisation rend ses performances équivalentes aux APT. En résumé, les algorithmes qui se comportent le mieux pour le MSE, avec ou sans fonction pénalisation, sont *convexos10* et *convex*.

LMH sur 80 itérations

Avec les figures 6a et 7a, il est possible de voir l'évolution de la LMH selon les itérations. Bien que la LMH semble être quasi indépendante du terme de régularisation, il converge vers des valeurs différentes, soit 2 et 5 voxels respectivement pour tous les algorithmes sauf les GC. D'ailleurs, le comportement des variantes du *GC* dans les 30 premières itérations est complètement différent de ceux des autres algorithmes. Les hypothèses retenues, après analyse des images à chaque 10 itérations, sont que la fonction barrière est trop forte au départ ou que sa force n'est pas suffisamment diminuée au fil des itérations. Ces hypothèses sont motivées par le fait que les premières estimées de ces algorithmes ont des coefficients d'atténuation élevés dans des zones où ils devraient être nuls. Pour être en mesure de départager les procédures selon la LMH, il faut regarder les figures 6b et 7b qui représentent un zoom des figures précédentes sur les meilleurs algorithmes. Tout d'abord, notez que *gradient* et *gradient_P* semblent être meilleur que leurs confrères. Toutefois, cette avance semble être favorisée par la limite de

11. Ce constat n'est pas vrai pour les variantes du gradient conjugué. Cette contradiction provient sûrement du fait qu'elles ne sont pas encore suffisamment près de leur solution.

80 itérations puisque *L-BFGS-B* et *L-BFGS-BFObj1* semblent être en mesure de les rattraper. De plus, les méthodes *convex* et *convexos10* n'ont pas eu le même genre de départ et *convexos10* a toujours un peu d'avance sur *convex*. Pour finir, les APT semblent tous avoir un comportement semblable, et ce, dans le cas avec et sans le terme de régularisation. Cette constatation est logique avec le fait qu'ils ont tous plus ou moins été construits sur les mêmes bases. En outre, les variantes de *L-BFGS-B* gèrent plus difficilement l'ajout d'un terme régulateur que les APT. Considérant la vitesse de convergence de ce dernier, il faudrait peut-être songer à diminuer la force de la fonction de pénalisation lorsqu'il a atteint une certaine valeur de la fonction objectif.

Variance sur 80 itérations

L'analyse de l'évolution de la variance de la zone B¹² selon les itérations confirme certaines des conclusions obtenues en observant le MSE. La figure 8a montre que lorsque les algorithmes itèrent trop longtemps, ils commencent à inclure du bruit dans la solution. De plus, elle appuie le fait que *L-BFGS-B* et *L-BFGS-BFObj1* débutent tôt à reproduire le bruit. La figure 9a, pour sa part, apporte un peu plus d'information. Ce graphique semble indiquer que les APT ont plus tendance que les SG à reconstruire le bruit localement, ce qui ne se voyait pas dans les graphiques de MSE. Pour finir, la figure 8b, qui est un zoom sur les lignes brisées les plus basses de la figure 8a, semble indiquer que les variantes du *GC* ont un contrôle acceptable du bruit localement lorsqu'il n'inclut pas un terme de régularisation. Cette dernière remarque supporte l'idée que ces algorithmes ont du potentiel et qu'une meilleure gestion de l'interaction entre la fonction barrière et de pénalisation, suivie d'une meilleure majoration, pourraient améliorer leurs performances.

5.2. Comparaison des solutions obtenues à l'arrêt Variance avec critère d'arrêt sur LMH

L'idée de cette sous-section est de présenter le meilleur algorithme lorsque le but est d'obtenir des solutions dont les zones uniformes ont le moins de bruit possible. Si les figures 10a et 10b sont utilisées pour comparer les méthodes, il en ressort que les algorithmes *trml_P* et *GC_P* sont les plus performants. En outre, les algorithmes avec un terme régulateur se comportent mieux que leur confrère. Toutefois, il faut savoir que *GC_P* a de la difficulté à atteindre le critère d'arrêt. Donc, il est possible qu'il n'ait simplement pas convergé suffisamment pour reconstruire le bruit. Dans une situation où utiliser un algorithme avec un terme de pénalisation est déconseillé, notez que l'algorithme *trml* est un choix judicieux.

MSE avec critère d'arrêt sur LMH

Maintenant, les algorithmes seront classés selon le MSE entre la solution à l'arrêt de l'algorithme et la mire originale. En observant les figures 11a et 11b, il est possible de constater que les résultats sont beaucoup plus stables avec les méthodes pénalisées, mais que les résultats dans les cas peu bruités sont meilleurs avec les procédures standard. Donc, si le niveau de bruit est relativement faible, il est avantageux de prendre une méthode standard. Plus particulièrement, la

12. Voir section 6.2

figure 11a montre qu'utiliser $L\text{-BFGS-}B$ ou $L\text{-BFGS-BFob1}$, lorsque les données sont prises dans des bonnes conditions, est un choix judicieux. Cependant, si des solutions stables sont recherchées, mieux vaut prendre des algorithmes pénalisés. En outre, $\text{convexos_}P$ a obtenu des résultats supérieurs et plus stables que ses confrères pénalisés. Toutefois, il ne faut pas écarter $L\text{-BFGS-}B_P$ et $L\text{-BFGS-BFob1_}P$ qui sont très proches de celui-ci. Afin d'appuyer les résultats précédents, la section 6.5 montre des images obtenues avec certains algorithmes pour la mire Shepp-Logan-easy lorsque la source envoie théoriquement 1000 photons. Les reconstructions obtenues par les algorithmes pénalisés sont flous ce qui indique que le gamma choisi était un peu trop fort.

Temps de calcul avec critère d'arrêt sur LMH

Le dernier critère de comparaison utilisé dans ce projet est la durée de reconstruction. Les données obtenues pour chacun des algorithmes sont illustrées dans la figure 12a. La figure montre que convexos se distingue des autres méthodes. Ensuite, il y a les variantes de $L\text{-BFGS-}B$ suivi de près par convex et Gradient . Notez que la durée affichée n'a pas été pénalisée lorsque la méthode est arrêtée avant d'atteindre la LMH visé. Toutefois, les variantes de GC sont celles qui ont le moins réussi à atteindre cette cible, ce qui ne change pas le classement actuel. $L\text{-BFGS-}B_P$ et $L\text{-BFGS-BFob1_}P$ ont eu, eux aussi, quelques problèmes à arriver à l'objectif. Donc, leurs valeurs sont sous-évaluées dans cette figure. La figure 12b n'inclut que les tests qui ont atteint la LMH critique. Cette figure montre seulement que les variantes du GC utilisées dans ce travail ont de la difficulté à converger et que les algorithmes pénalisés n'atteignent pas toujours le critère d'arrêt. Cette dernière remarque pourrait provenir de la force du gamma.

Conclusion

Pour terminer, un rappel sur certains des résultats obtenus. Pour commencer, l'utilisation de méthodes standards devrait être limitée aux données qui sont peu contaminées par du bruit puisque la qualité de la reconstruction est très affectée par la présence de bruit. Sinon il vaut mieux utiliser les méthodes robustes que sont les algorithmes pénalisés. De plus, malgré les réserves faites sur la convergence des APT utilisant des OS, les figures de ce projet semblent indiquer que ces algorithmes ont un comportement égal ou même supérieur aux autres méthodes. Ensuite, les résultats de ce projet démontrent encore que les variantes de $L\text{-BFGS-}B$ sont très efficaces en tomodensitométrie. Toutefois, les résultats de ce document déconseillent de l'utiliser sur des données relativement bruitées sans terme de pénalisation. Enfin, les résultats de l'implantation de variantes de gradient conjugué n'ont pas été fructueux. Or, ils indiquent que cette méthode possède du potentiel. Il suffirait de la retravailler sur les points notés précédemment. Ce document présente une comparaison entre quelques algorithmes applicables à la tomodensitométrie. Il est possible d'approfondir le sujet de plusieurs façons. Quelques unes sont présentées dans les articles [FR96], [HGD11] et [TTLD11].

6. Annexe

6.1. Notations.

	Nombre de projections	multiplié par le nombre de détecteur	()
	Nombre de voxels	()	
	Vecteur de	éléments tel que	est la moyenne d'absorption dans le voxel
	Matrice	tel que	est la longueur d'intersection entre la projection et le pixel
	Indice utilisé pour représenter les voxels traversés par une projection		
	Vecteur colonne de	éléments tel que	est le nombre de photons détecté à la projection lorsque l'objet est présent
	Indice utilisé pour représenter les différents détecteurs		
	Vecteur colonne de	éléments tel que	où
	Constante utilisée dans la fonction pénalisation		
	Si et partagent 0, 1 ou 2 vertex, alors	vaut respectivement	, et
	Ensemble de valeurs tel que	représente le poids relatif du voxel	(Définit en [FR96])
	Constante utilisée dans la fonction potentiel		
	Utilisée pour évaluer	, elle modifie la force de la fonction potentiel	
	Utilisée pour évaluer	, elle modifie le comportement de la fonction	

6.2. Zone d'importance pour l'évaluation du LMH et de la variance

L'image qui suit est la superposition des 4 mires à laquelle a été ajouté la position de la zone A et celle de la ligne B.

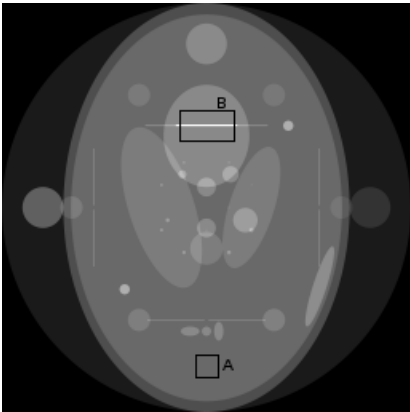


FIGURE 1. Zone utilisée pour les tests

6.3. Configuration d'acquisition.

Dimension de l'image	cm x cm
Discrétisation	x
Mode d'acquisition	Géométrie à angles égaux
Nombre de détecteurs	
Nombre de projections	
Largeur angulaire du panneau des détecteurs	
Distance entre la source et le point de rotation	cm
Rayon de courbure des détecteurs	cm

6.4. Mires.

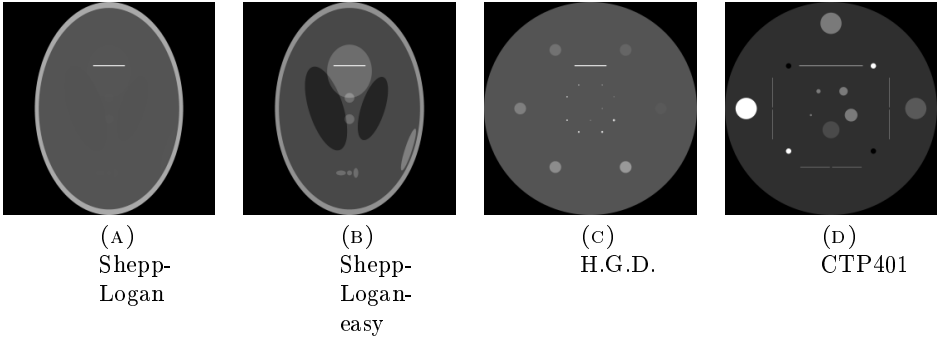


FIGURE 2. Mires : Noir et blanc

6.5. Comparaison d'images obtenues avec quelques algorithmes

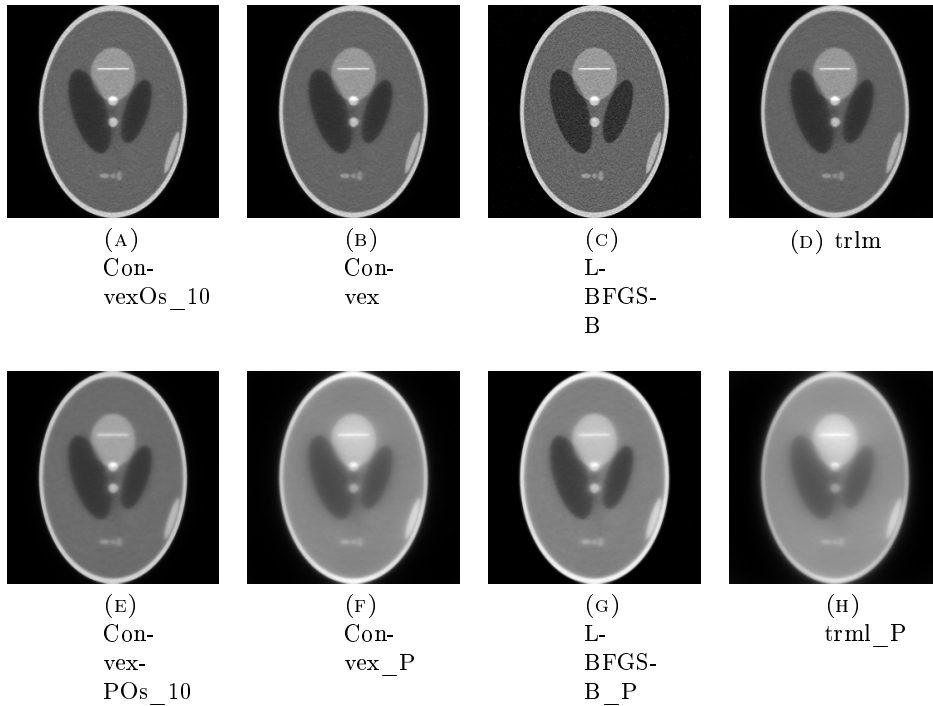


FIGURE 3. Images obtenues pour différents algorithmes avec la mire Shepp-Logan-easy et en utilisant une source de 1000 photons. Les algorithmes standards et pénalisés ont été arrêtés lorsque la ligne B de la solution courante avait un LMH de 2 et de 5 respectivement. Un pixel noir et un pixel blanc représentent respectivement un coefficient d'atténuation de et de

6.6. Figures utilisées dans l'analyse

Soit MSE_k le MSE entre la mire et la solution à l'itération k d'un algorithme appliqué sur un sinogramme de la mire avec un niveau de bruit σ . Les mires et les niveaux de bruit sont introduits à la section 3. Soit Var_k et LMH_k respectivement la variance de la zone A et la largeur à mi-hauteur de la ligne B (Voir section 6.2) de la solution à l'itération k d'un algorithme appliqué sur un sinogramme de la mire avec un niveau de bruit σ . Alors, MSE , Var et LMH sont respectivement les moyennes des MSE_k , des Var_k et des LMH_k pour les 4 mires avec chacun des 4 niveaux de bruit.

Soit MSE_{Recon} et Var_{Recon} respectivement MSE_k et Var_k où k représente l'itération à laquelle la solution courante a atteint le critère d'arrêt. Soit $Recon_Time$ le temps nécessaire pour atteindre le critère d'arrêt d'un algorithme appliqué sur un sinogramme de la mire avec un niveau de bruit σ . Alors, MSE et Var sont respectivement la moyenne des MSE_{Recon} et la moyenne des Var_{Recon} sur les 4 mires à un niveau de bruit σ . L'ordre, de gauche à droite, des niveaux de bruit est noNoise, 10000c, 1000c et 100c. $Recon_Time$ est la moyenne des $Recon_Time$ pour les 4 mires avec chacun des 4 niveaux de bruit.

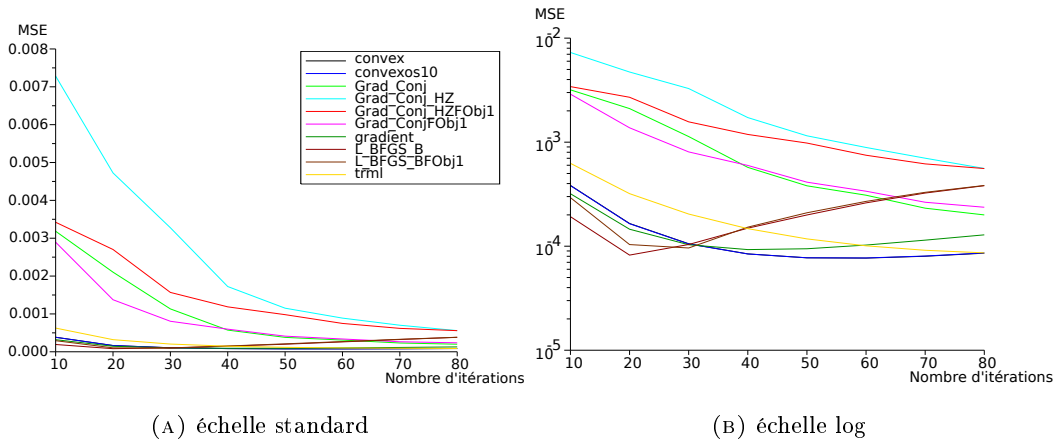


FIGURE 4. MSE à chaque 10 itérations pour chacun des algorithmes standards

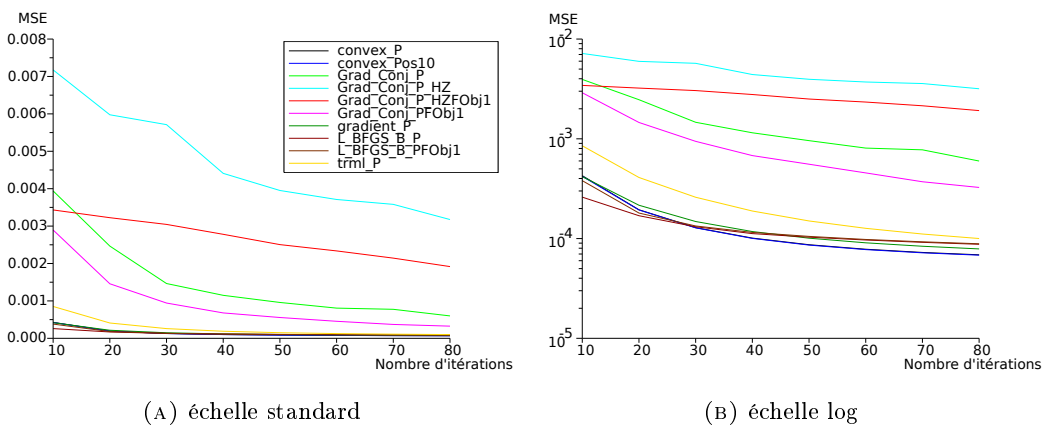


FIGURE 5. MSE à chaque 10 itérations pour chacun des algorithmes pénalisés.

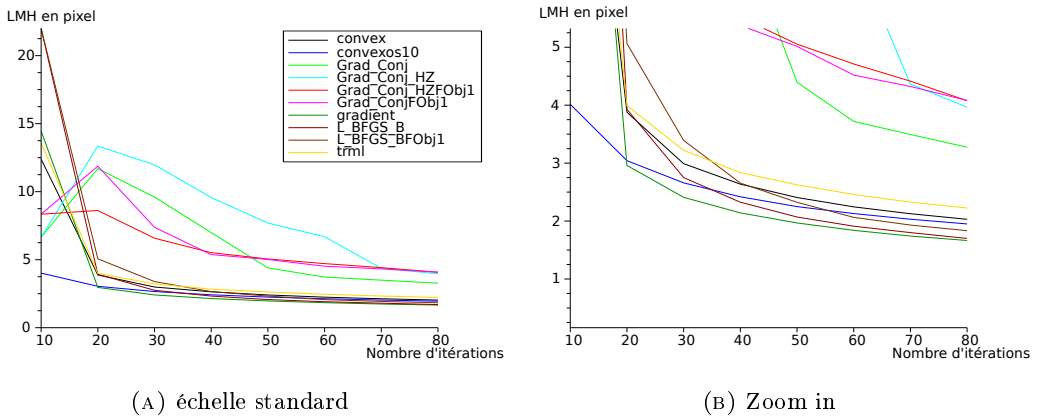


FIGURE 6. LMH à chaque 10 itérations pour chacun des algorithmes standards

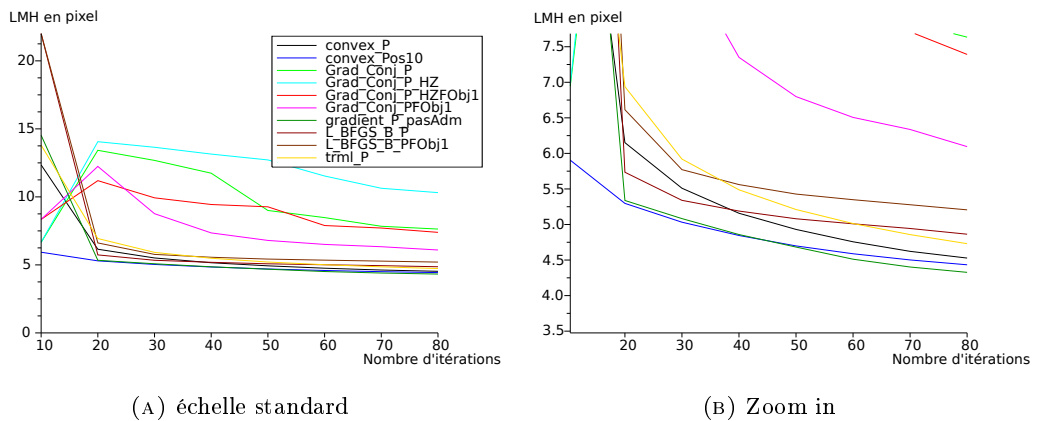


FIGURE 7. LMH à chaque 10 itérations pour chacun des algorithmes pénalisés

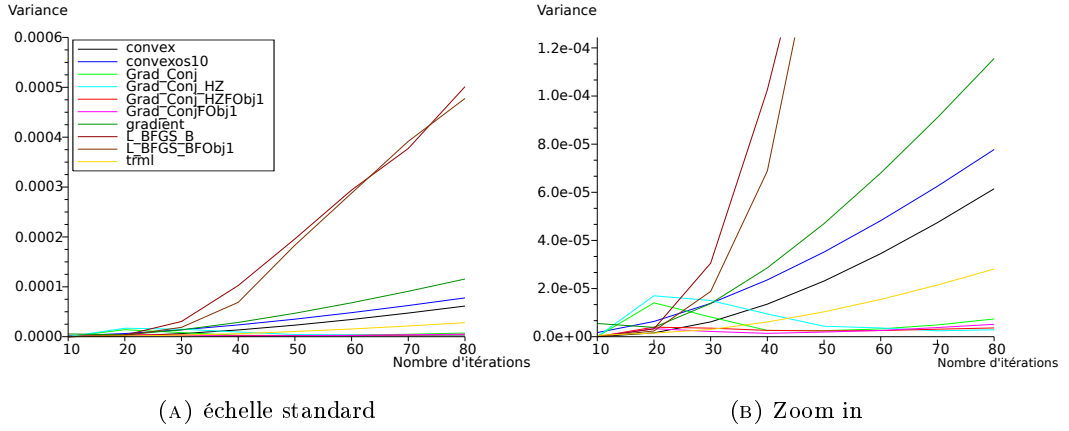


FIGURE 8. Var à chaque 10 itérations pour chacun des algorithmes standards

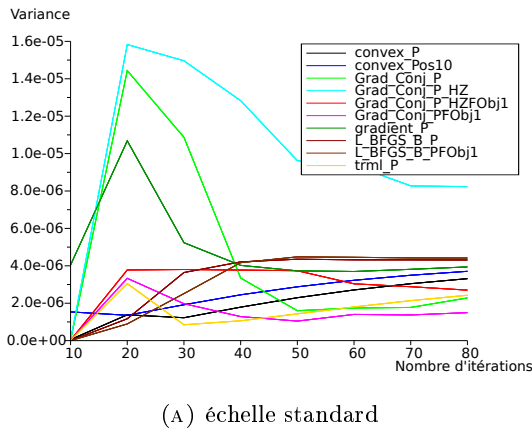
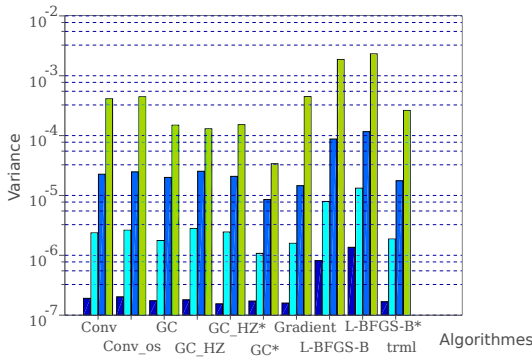
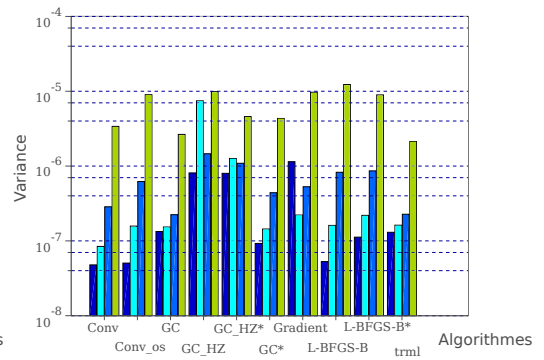


FIGURE 9. Var à chaque 10 itérations pour chacun des algorithmes pénalisés

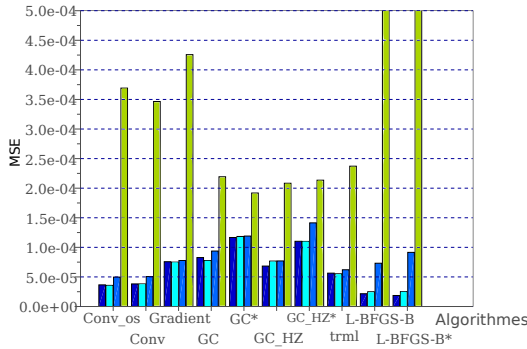


(A) LMH critique : 2

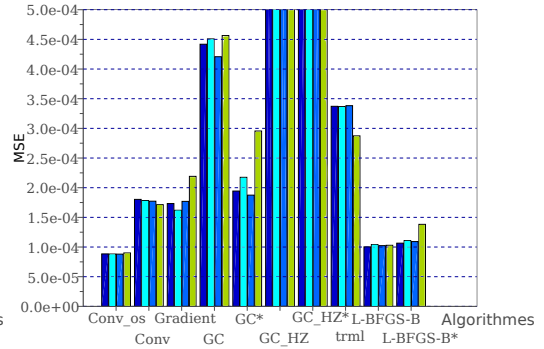


(B) LMH critique : 5

FIGURE 10. Var selon les algorithmes pénalisés et le bruit représenté avec une échelle log. Les algorithmes sont arrêtés lorsque la solution courante a atteint la LMH critique ou lorsque le nombre d'itérations dépasse 200. Les couleurs de gauche à droite représentent respectivement noNoise, 10000c, 1000c et 100c.



(A) LMH critique : 2



(B) LMH critique : 5

FIGURE 11. MSE selon les algorithmes pénalisés et le bruit représenté avec une échelle log. Les algorithmes sont arrêtés lorsque la solution courante a atteint la LMH critique ou lorsque le nombre d'itérations dépasse 200. Les couleurs de gauche à droite représentent respectivement noNoise, 10000c, 1000c et 100c.

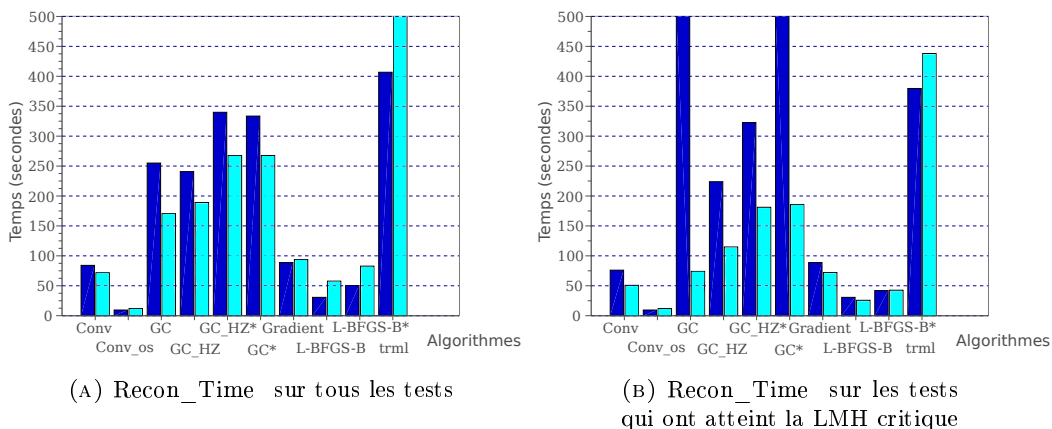


FIGURE 12. Recon_Time selon les algorithmes et l'inclusion d'un terme de régularisation. Les algorithmes ont été arrêtés lorsque la LMH de la solution courante atteint la valeur critique ou lorsque le nombre d'itérations dépasse 200. La valeur critique pour les algorithmes standards(Bleu) et les algorithmes pénalisés(Cyan) sont respectivement de 2 et de 5. L'absence d'une colonne indique qu'aucun des tests a convergé.

Références

- [CMI09] É. Chouzenoux, S. Moussaoui & J. Idier, "A majorize-minimize line search algorithm for barrier function", version 4, 31 p., 2009.
- [D95] J-P. Dussault, "Numerical stability and efficiency of penalty algorithms", SIAM Journal on Numerical Analysis, Vol 32, No. 1, pp. 296-317, 1995.
- [FR96] J.A. Fessler & W.L. Rogers, "Resolution properties of regularized image reconstruction methods", Technical Report No. 297, 25 p., 1996.
- [H80] G.T. Herman, "Image reconstruction from projections : The fundamentals of computerized tomography", IEEE Transactions on Signal Processing, vol. 41, no 2, pp. 534-548, 1980.
- [HGD11] B. Hamelin, Y. Goussard & J-P. Dussault, "Comparison of optimization techniques for regularized statistical reconstruction in x-ray tomography", IEEE IPTA, pp. 87-91, 2011.
- [HZ05] W.H. Hager & H. Zhang, "A survey of nonlinear conjugate gradient methods", 21 p., 2005.
- [KB05] J.S. Kole & F.J. Beekman, "Evaluation of the ordered subset convex algorithm for cone-beam CT", Physics in Medicine and Biology, pp. 613-623, 2005.
- [KS88] A.C. Kak & M. Slaney, "Principles of Computerized Tomographic Imaging", IEEE, 327 p., 1988.
- [LC84] K. Lange & R. Carson, "EM reconstruction algorithms for emission and transmission tomography", Journal of Computer Assisted Tomography pp. 306-316 1984.
- [LF95] K. Lange & J.A. Fessler, "Globally convergent algorithms for maximum a posteriori transmission tomography", IEEE Trans on Image Processing pp. 1430-1438, 1995.

- [RP86] J. Radon & P.C. Parks, (translator), "On the determination of functions from their integral values along certain manifolds", IEEE Transactions on Medical Imaging 5, pp. 170–176, 1986.
- [SB93] K. Sauer, & C. Bouman , "A local update strategy for iterative reconstruction from projections," IEEE Transactions on Signal Processing, Vol. 41, no. 2, pp. 534-548, Feb. 1993.
- [TTLD11] M. Toussaint, C. Thibaudeau, R. Lecomte, J.P. Dussault, "Description théo., analyse et comparaison de procédures d'optim. en tomodensitométrie.", Rapport de stage, U. Sherbrooke, 30 p., 2011.

MAXIME TOUSSAINT, DÉPARTEMENT DE MATHÉMATIQUE DE L'UNIVERSITÉ DE SHERBROOKE

CHRISTIAN THIBAudeau, DÉPARTEMENT DE GÉNIE ÉLECTRIQUE ET INFORMATIQUE DE L'UNIVERSITÉ DE SHERBROOKE.

Courriel: `Maxime.Toussaint@usherbrooke.ca`

Courriel: `Christian.Thibaudeau@usherbrooke.ca`

GRUPE DE TRESSES ET LES SURFACES HYPERELLIPTIQUES

FÉLIX CAMIRAND

RÉSUMÉ.

Etant donné une courbe hyperelliptique

où g ou $g+1$, on y associe un revêtement ramifié

à deux feuillets avec $2g+2$ points de ramifications. Le théorème d’Arnold [1] stipule que l’action du groupe de tresses sur le groupe d’homologie coïncide avec le groupe des matrices symplectiques $Sp(2g, \mathbb{Z})$ seulement dans les cas où g est pair, $g=1$ et $g=2$. En ce qui a trait au cas où g est différent de 1 et 2, il existe au moins une matrice qui ne peut être représentée comme l’action du groupe de tresses sur le groupe d’homologie $H_1(X, \mathbb{Z})$. Dans cet article, nous allons montrer que l’action du groupe de tresses sur le groupe d’homologie $H_1(X, \mathbb{Z})$ correspond à un sous-groupe du groupe des matrices symplectiques $Sp(2g, \mathbb{Z})$.

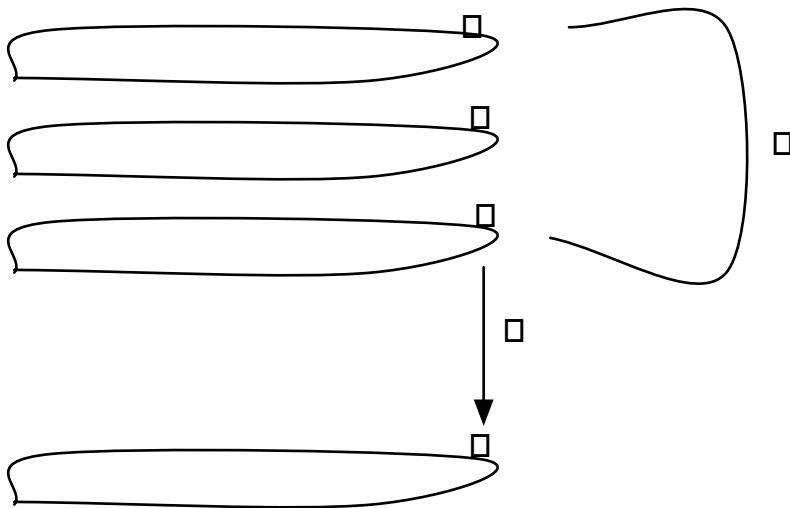
1. Revêtement Ramifié

Dans cette section, nous allons introduire une notion essentielle à la compréhension de cet article, soit celle de revêtement ramifié. Soient X et Y , deux espaces topologiques. Un revêtement

est une application continue et surjective tel que, pour tout point y dans Y , il existe un ouvert U contenant y tel que $\pi^{-1}(U)$ est une réunion disjointe d’ensembles ouverts dans X , réunion dans laquelle chaque composante de $\pi^{-1}(U)$ est homéomorphe à U , et l’homéomorphisme est établi par π .

Je tiens à remercier Vasilisa Shramchenko qui a su m’épauler et me soutenir tout au long de ce processus de recherche. Ses judicieuses remarques ont brillamment éclairé ce long chemin parfois sinueux qui a mené à la rédaction de cet article. Je tiens à remercier aussi toute l’équipe du CAMUS sans qui cet ouvrage n’aurait pas été possible.

FIGURE 1. Exemple de Revêtement



Par exemple, si Y est un espace topologique, que X se compose de trois copies de Y , et que p est la projection de Y sur X , alors

p est un revêtement, comme on peut le voir à la figure 1.

Soient Y et X , deux surfaces connexes orientables et compactes. On dit que

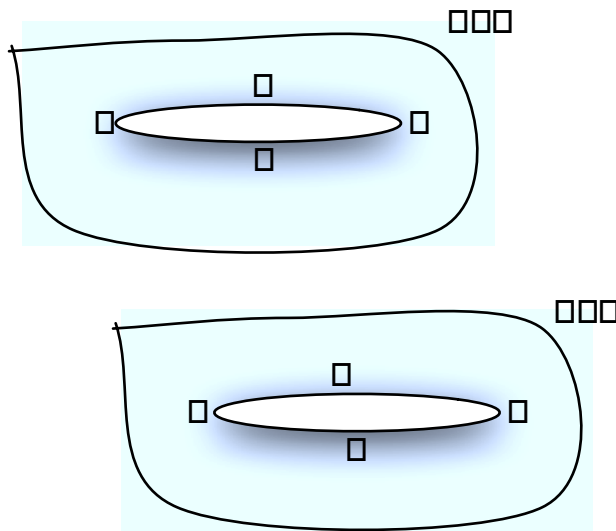
p est un revêtement ramifié si :

- (1) p est continu, surjectif et préserve l'orientation.
- (2) Il existe un ensemble fini de points de X , R , tel que $p^{-1}(X \setminus R)$ soit un revêtement de $X \setminus R$.
- (3) Pour tout $x \in X \setminus R$, où x varie entre 0 et 1 , il existe un voisinage U dans lequel se trouve x et une coordonnée complexe z dans U , et pour tout point dans U , il existe une coordonnée complexe z tel que

dans le voisinage de ce point de X .

Il est à noter qu'il y a toujours un nombre fini de points de ramification pour Y et X compactes. Tous les points qui sont réguliers, c'est-à-dire qui ne sont pas des points de ramification, ont le même nombre de pré-images sous p . Ce nombre est appelé le degré du revêtement. Les points de ramification sont les points qui comporte moins de pré-image que les autres. Dans cet article, nous nous intéresserons à un type de revêtements ramifiés dans lequel les points réguliers n'ont que deux pré-images. Nous verrons dans la prochaine section un exemple qui rendra ce concept plus clair.

FIGURE 3. Construction de la Courbe hyperelliptique



Alors,

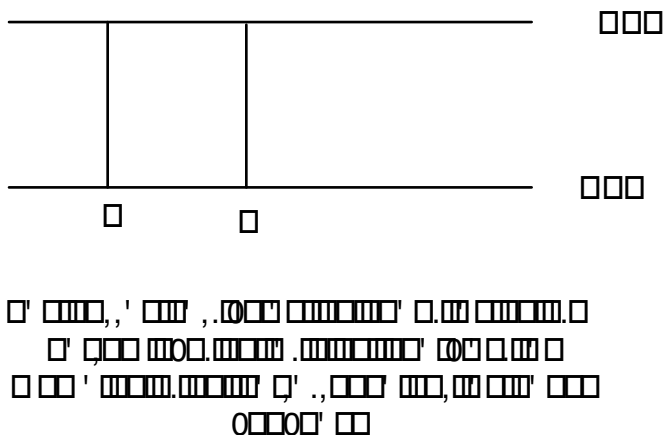
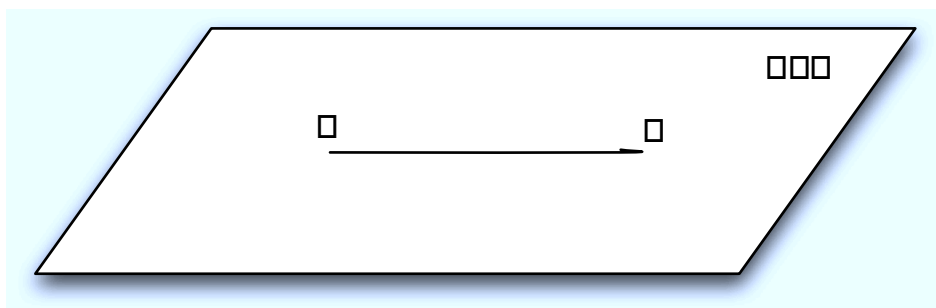
$$- \quad - \quad -$$

et on a une courbe

Ainsi, on voit bien que pour chaque valeur de λ , il y a deux valeurs de μ possibles, sauf dans le cas de $\lambda = 0$ et $\lambda = \infty$. Ce sont les points de ramification du revêtement

C'est pourquoi, si on souhaite tracer le graphe dans $\lambda - \mu$, il faut avoir recours à deux couches.

L'idée consiste à prendre deux copies de \mathbb{P}^1 , et d'effectuer une coupure entre 0 et l'infini sur chacune des copies, pour ensuite séparer et identifier les deux rives, pour enfin coller ensemble les rives opposées, comme indiqué sur la figure 3. On a alors, entre les deux copies de \mathbb{P}^1 , une structure en spirale, et on passe d'une couche à l'autre lorsque λ va de 0 à ∞ . On peut représenter la courbe hyperelliptique de plusieurs manières, dont deux nous seront particulièrement utiles, soit le diagramme de Hurwitz (figures 4 et 6) et la représentation vue du dessus (figures 5 et 7). Dans la représentation par le diagramme de Hurwitz, chaque étage représente une couche, et on passe d'une couche à une autre en passant par la coupure entre 0 et l'infini. Les barres verticales sont les points de ramification des deux couches, soit les points qui se trouvent sur deux couches en même temps.

FIGURE 4. Diagramme de Hurwitz pour $n=1$,FIGURE 5. Représentation Vue du Dessus pour $n=1$ 

Maintenant, supposons que l'on a une courbe hyperelliptique de la forme

avec g entier. Dans ce cas, on voit qu'il y a $2g+2$ points de ramifications de type «fini», qui sont $\pm \sqrt{a}$ et $\pm \sqrt{b}$. Le degré du polynôme en y est impair. Il y a donc un point de ramification à l'infini. Si on a une courbe hyperelliptique de la forme

avec g entier, alors on a $2g$ points de ramification de type «fini». Il n'y a aucun point de ramification à l'infini. Ainsi, dans les deux cas, on a toujours $2g+2$ points de ramification.

La manière de construire un revêtement ramifié à $2g+2$ points de ramification est la suivante :

FIGURE 6. Diagramme de Hurwitz pour

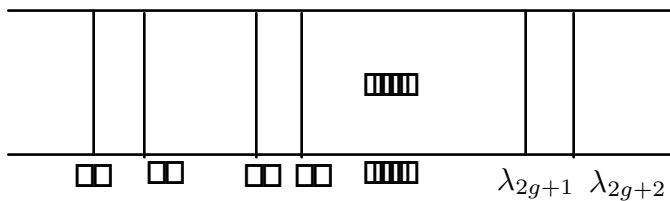
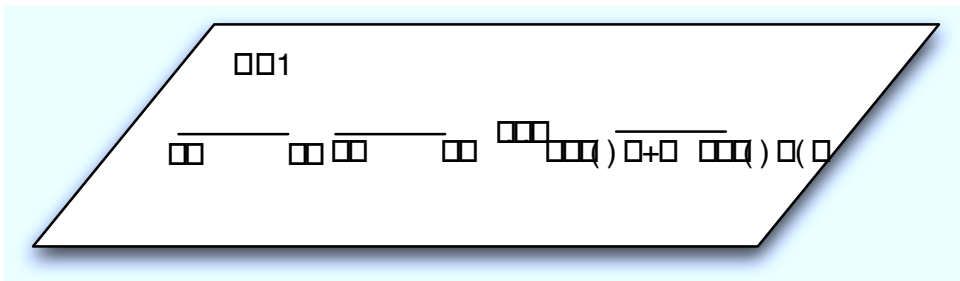


FIGURE 7. Représentation Vue du Dessus pour



- (1) On forme couples de points. Chaque point doit se retrouver dans un et un seul couple.
- (2) On procède comme dans le cas précédent : On se prend d'abord deux copies de .
- (3) Pour chacun des couples de points, on coupe entre les deux points du couple, on identifie les deux rives, et on colle les rives opposées ensemble.

3. Groupe Symplectique

Une matrice de dimension à coefficient dans est dite symplectique si elle préserve les formes bilinéaires antisymétriques, ou plus simplement, si elle respecte la condition suivante :

Dans cette section, nous étudierons quelques propriétés des matrices symplectiques, puisque c'est ce type de matrice que nous allons utiliser pour caractériser l'action du groupe de tresses (que nous allons définir à la prochaine section) sur les points de ramification d'une surface hyperelliptique. On remarque ainsi que si A est une matrice symplectique, alors A est inversible, et son inverse est donnée par

qui est aussi une matrice symplectique à coefficient dans \mathbb{R} . Enfin, si A et B sont des matrices symplectiques de dimension $2g$ à coefficient dans \mathbb{R} , on a que

et

et ainsi

On en conclue donc que le produit de deux matrices symplectiques est aussi une matrice symplectique, ce qui confère à l'ensemble des matrices symplectiques une structure de groupe avec le produit matriciel usuel. Tout au long de cet article, le groupe symplectique se constituant des matrices de dimension $2g$ à coefficient dans \mathbb{R} sera noté $\mathrm{Sp}(2g, \mathbb{R})$.

Soit A , une matrice de $\mathrm{Sp}(2g, \mathbb{R})$, telle que

où J , S , et T sont des matrices de dimension $2g$ à coefficient dans \mathbb{R} . S est une matrice de $\mathrm{Sp}(2g, \mathbb{R})$, ce qui équivaut à écrire que

et donc que

ou plus simplement

On peut montrer que, pour g supérieur ou égal à 2, le groupe symplectique est engendré par trois types de matrices.

(1) Les matrices de translation. Ces matrices sont de la forme

où x est un réel. Soient deux matrices de translation T_x et T_y tel que

et

Alors, on a que

On en déduit aisément que les matrices de translation forment un groupe pour la multiplication matricielle usuelle.

(2) Les matrices de rotation. Ces matrices sont de la forme

avec A une matrice de dimension n et à coefficient dans \mathbb{C} tel que le déterminant de A soit égal à 1. Soient deux matrices de rotation R_1 et R_2 tel que :

et

Alors

Il est alors clair que les matrices de rotation forment un groupe pour la multiplication matricielle usuelle.

(3) Les matrices semi-involutives. Ces matrices sont de la forme

où D est une matrice diagonale avec les éléments diagonaux égaux à 0 ou à 1.

Regardons plus en détail le cas dans lequel $n=2$, puisque c'est le cas qui nous intéresse dans cet article. Le groupe de translation est alors engendré par trois matrices :

On sait déjà que le groupe des matrices unimodulaires ayant deux lignes et deux colonnes à coefficient dans \mathbb{C} peut être engendré par trois matrices :

On en déduit simplement que le groupe des matrices de rotation est engendré par trois matrices :

Enfin, pour $g=2$, il existe seulement trois matrices semi-involutives, qui sont l'identité, et

En mettant tout cela ensemble, on remarque que :

—
—
—
—

On en conclue donc que le groupe symplectique est engendré par ces quatre matrices, soit et . Il est possible de montrer que ces quatre matrices sont aussi indépendantes.

4. Groupe de tresses

On appelle le groupe de tresses à brins le groupe engendré par n générateurs soumis aux relations suivantes : Pour tout i variant entre 1 et $n-1$,

(1) Si $\sigma_i^2 = 1$, alors

(2) Si $[\sigma_i, \sigma_{i+1}] = 1$, alors

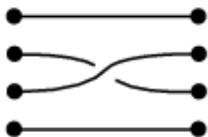
On visualise ce groupe comme suit. Prenons n points à gauche et n points à droite, reliés par n brins, chacun des brins ayant un point à gauche et un point à droite comme extrémité. Le générateur σ_i agit sur les points i et $i+1$ en les échangeant. La tresse σ_i agit elle aussi sur les points i et $i+1$.

Par exemple, si on considère le groupe de tresses B_3 , on se retrouve avec avec trois générateurs :

(1) :



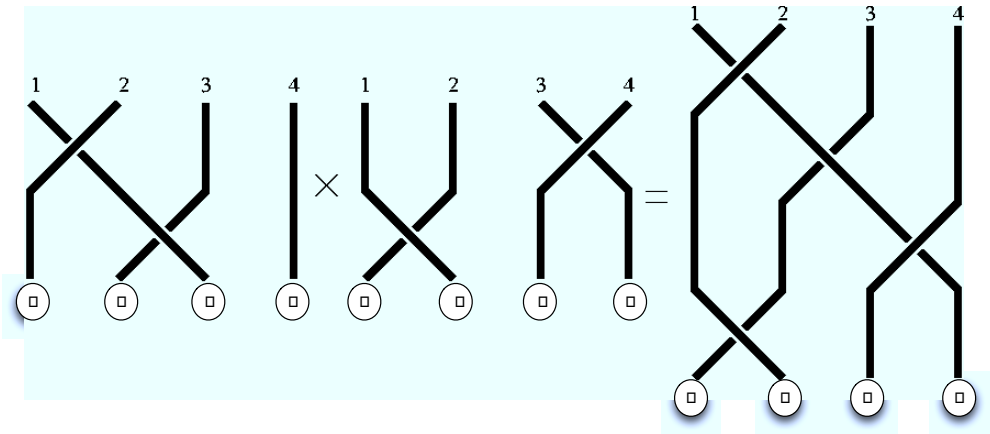
(2) :



(3) :



FIGURE 8. Produit de tresses



est alors représenté de la manière suivante :



L'opération de ce groupe est appelée produit de tresses, et s'effectue de la manière suivante : étant donné deux tresses σ et τ , le produit de σ et τ est obtenu en concaténant les deux tresses, c'est-à-dire en ajoutant la tresse τ à la suite de la tresse σ , comme on peut le voir à la figure 8.

5. Groupe d'homologie

Avant de poursuivre, nous allons définir une notion qui nous sera essentielle pour la compréhension du théorème d'Arnold, celle du groupe d'homologie. Dans le but de pouvoir définir clairement ce qu'est le groupe d'homologie, nous allons revenir sur quelques notions fondamentales. La première de ces notions est celle d'homotopie. Étant donné deux chemins continus γ_1 et γ_2 ayant le même point de départ ainsi que le même point d'arrivée dans un espace topologique X , on dit que γ_1 et γ_2 sont homotopes s'il est possible de déformer l'un pour obtenir l'autre de manière continue (voir figure 9). En d'autres termes, si on a :

avec γ_1 et γ_2 , alors γ_1 et γ_2 sont homotopes s'il existe une fonction continue

tel que

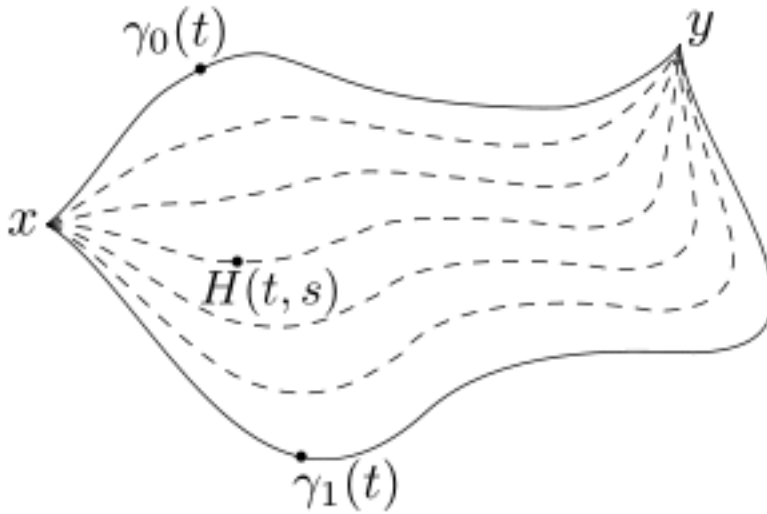
pour tout

et

et

pour tout

FIGURE 9. Homotopie



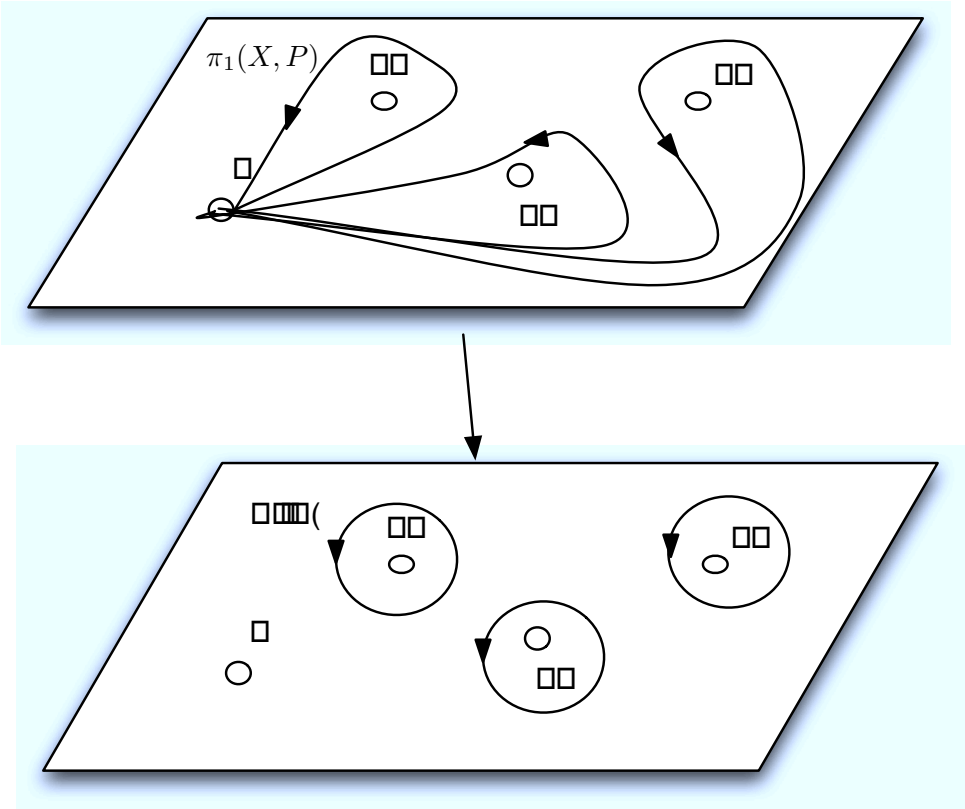
L'homotopie est une relation d'équivalence. Maintenant, passons à la notion de groupe fondamental. Étant donné un espace topologique X et un point $x_0 \in X$, le groupe fondamental basé en x_0 , noté $\pi_1(X, x_0)$, est l'ensemble des classes d'équivalence des chemins orientés dans X débutant et se terminant en x_0 . Cet ensemble est un groupe pour la composition de chemins. Pour X connexe,

Enfin, on appelle le premier groupe d'homologie $H_1(X, \mathbb{Z})$ le groupe fondamental factorisé par la relation

Prenons un exemple. Soit $X = \mathbb{R}^2$ et prenons $x_0 = (0, 0)$ dans X . On peut voir la comparaison entre le groupe fondamental $\pi_1(X, x_0)$ et le groupe d'homologie $H_1(X, \mathbb{Z})$ à l'aide des générateurs de ces groupes dans la figure 10.

Pour conclure cette section, nous allons introduire la notion d'indice d'intersection de deux chemins à l'aide de l'exemple illustré de la figure 11. Étant donné deux chemins γ et η , et que γ et η ne se croisent qu'une seule fois, et que l'on souhaite connaître l'indice d'intersection $I(\gamma, \eta)$, alors, on s'intéresse au repère cartésien formé par γ et η dans le voisinage de l'intersection. Si ce repère est positif, alors l'indice d'intersection $I(\gamma, \eta)$ vaut 1. Sinon, il vaut -1 . Si γ et η se croisent plus d'une fois, alors l'indice d'intersection $I(\gamma, \eta)$ a la valeur de la somme sur toutes les intersection entre γ et η des indices d'intersection obtenus de la manière dont on vient de la décrire. De plus, on dira que $I(\gamma, \eta)$ vaut $-I(\eta, \gamma)$. Ainsi, si $I(\gamma, \eta) = 1$, alors $I(\eta, \gamma) = -1$ et ainsi l'indice d'intersection est une forme bilinéaire antisymétrique. Nous soulignerons plus tard que c'est la raison pour laquelle les matrices symplectiques apparaissent.

FIGURE 10. Générateurs du groupe fondamental et Générateurs du groupe d'homologie



6. Application de monodromie d'un revêtement ramifié

Si on fixe n points distincts x_1, \dots, x_n de X et que l'on note $\pi_1(X, P)$, choisissons un point P dans notre nouvel ensemble $X \setminus \{x_1, \dots, x_n\}$. Ensuite, supposons que l'on ait un revêtement ramifié

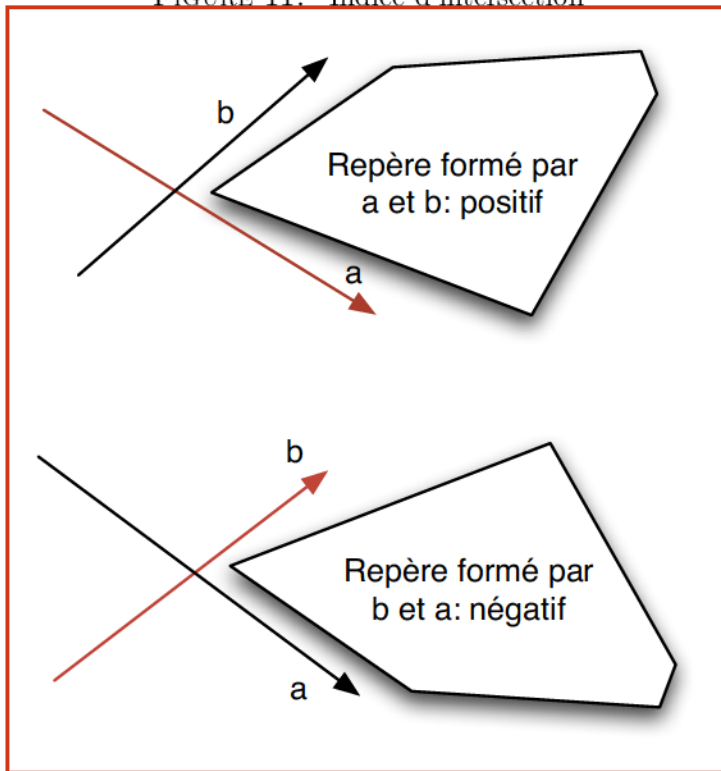
branché aux points x_1, \dots, x_n . Enfin, soit

le revêtement induit, c'est-à-dire le revêtement tel que, pour tout q dans X , on a

Si γ est une boucle orientée basée en Q et x est un point dans la préimage de Q sous π , alors il existe un unique chemin $\tilde{\gamma}$ dans \tilde{X} qui commence en x se terminant nécessairement dans $\pi^{-1}(x)$, et tel que $\pi \circ \tilde{\gamma} = \gamma$.

Si on note le point d'arrivée de $\tilde{\gamma}$ par \tilde{x} , on a alors une fonction

FIGURE 11. Indice d'intersection



$$P \mapsto \mu(\bar{\gamma}_P).$$

On voit tout de suite que $h_{\mu(\bar{\gamma}_P^{-1})} = h_{\mu(\bar{\gamma}_P)}^{-1}$, car $\bar{\gamma}_P^{-1}$ n'est rien d'autre que le chemin $\bar{\gamma}_P$ parcouru dans le sens contraire, donc le point d'arrivée devient le point de départ, et réciproquement. Tout ceci induit un homomorphisme

$$h_\mu : \pi_1(\widehat{\mathbb{CP}^1}, Q) \rightarrow S_d.$$

qui associe à une boucle orientée de $\pi_1(\widehat{\mathbb{CP}^1}, Q)$ une permutation $\sigma = h_\mu(\gamma)$ de S_d tel que si on numérote les points P_1, \dots, P_d de $f^{-1}(Q)$, et que σ permute les points de cet ensemble, alors $\sigma(P_i) = \mu(\bar{\gamma}_{P_i})$. On appelle cette application l'application de monodromie.

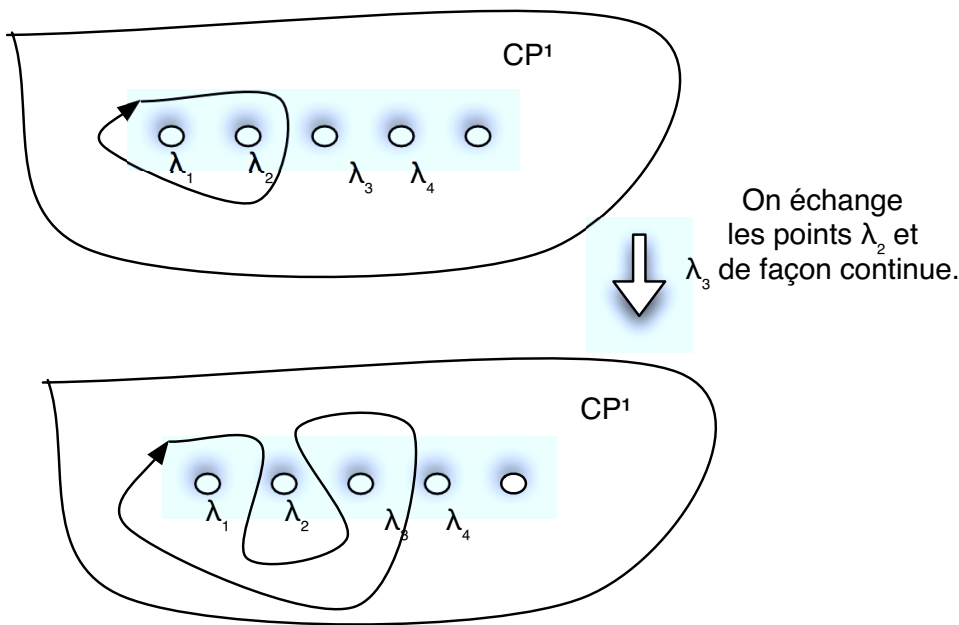
Voici un résultat intéressant qui nous permettra de mieux voir le groupe d'homologie sur $\widehat{\mathbb{CP}^1}$. Il existe $\gamma_1, \dots, \gamma_r$ dans $\widehat{\mathbb{CP}^1}$ basés en Q tel que

- (1) $\gamma_1, \dots, \gamma_r$ génèrent $\pi_1(\widehat{\mathbb{CP}^1}, Q)$
- (2) La seule relation entre les γ_i est $\gamma_1 \cdots \gamma_r = Id$
- (3) Chacun des γ_i est homotope à une petite boucle autour d'un Q_i .

Étant donné F , une courbe hyperelliptique de la forme

$$\mu^2 = \prod_{i=1}^n (\lambda - \lambda_i),$$

FIGURE 12. Exemple d'action de groupe de tresses



soit

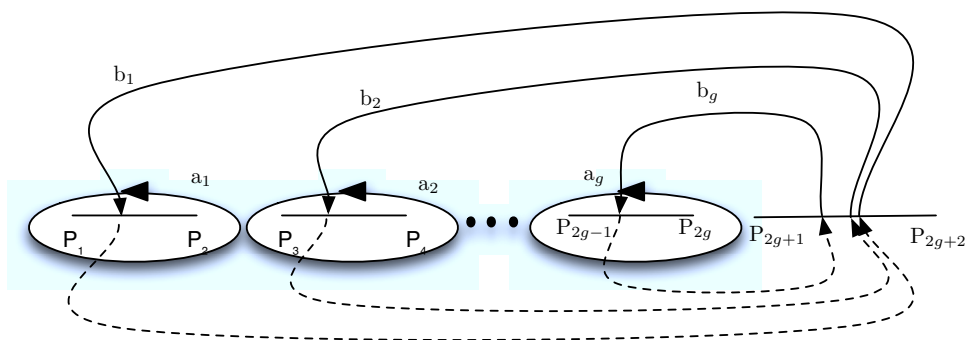
un revêtement ramifié y étant associé. Dans le cas d’une courbe hyperelliptique, l’action du groupe de tresses peut être représentée de la manière suivante. On imagine que l’on permute deux points de ramification fini de manière continue. Il est interdit de bouger le point de ramification à l’infini, lorsque g est impair (il n’y a pas de point de ramification à l’infini lorsque g est pair). Alors, les boucles orientées qui entouraient l’un ou l’autre des points permutés «subissent» le même sort que les points, comme l’indique l’illustration de la figure 12, dans laquelle on a appliqué le générateur $\sigma_{2,3}$ du groupe de tresses. Aussi, à toute tresse est associée une transformation linéaire dans $H^1(X, \mathbb{C})$. De la même manière, à chaque tresse, on associe une transformation linéaire dans le groupe d’homologie de la courbe F . Choisissons un ensemble de générateurs dans B_n . Une transformation linéaire dans $H^1(X, \mathbb{C})$ est donc représentée par une matrice de dimensions $2g \times 2g$. Appelons la matrice correspondant à une tresse M .

Le groupe d’homologie $H_1(X, \mathbb{Z})$ de la surface de genre g a $2g$ générateurs libres tel que leurs indices d’intersections sont :

- $\langle a_i, a_i \rangle = 0$ si
- $\langle a_i, b_i \rangle = 1$ si
- $\langle b_i, a_i \rangle = -1$ si

Ces générateurs sont appelés les générateurs canoniques.

FIGURE 13. Base Canonique



Les lignes pointillées sont sur l'autre feuillet.

Puisque l'indice d'intersection est une forme bilinéaire antisymétrique à coefficients entiers, et que les automorphismes σ préservent l'indice d'intersection, alors les matrices M_{σ} dans la base $\{a_i, b_i\}$ appartiennent au groupe symplectique. C'est pour cette raison que l'on en arrive à la représentation du groupe de tresses au sein du groupe symplectique par l'entremise de l'application

avec $\sigma(a_i) = a_i$ ou $\sigma(a_i) = b_i$.

7. Théorème d'Arnold

Maintenant, on peut reformuler le Théorème d'Arnold de la manière suivante : Pour $n \geq 3$, où $n \equiv 1 \pmod{2}$ et où $n \not\equiv 3 \pmod{4}$, et seulement pour ces valeurs de n , on a

La démonstration du fait qu'il n'y ait pas d'autres valeurs de n pour lesquelles

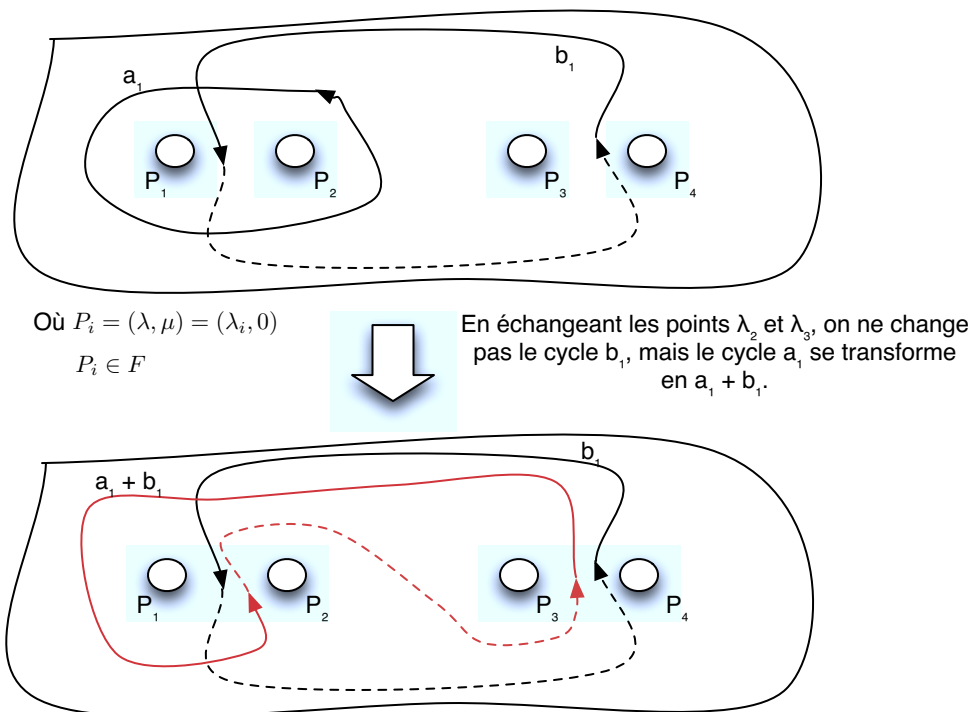
est en soi assez simple. Soit la base canonique représentée par le diagramme de la figure 13.

Soit $n \geq 3$. Si n n'est pas égal à 6, alors le cycle σ entoure deux points alors que le cycle τ entoure 4 points. Le groupe symplectique contient une transformation qui amène le cycle σ sur τ . Or, les automorphismes conservant le nombre d'éléments entourés par les cycles, il n'existe pas de transformation σ qui amène σ sur τ . On peut donc en conclure que, pour n différent de 3, 4 et 6, le groupe symplectique n'est pas égal à $Sp(2n, \mathbb{Z})$.

En ce qui a trait à la démonstration de la première partie du théorème, il s'agit d'une simple vérification. Pour $n = 3$ ou $n = 4$, on a que

sont des transformations de \mathcal{S}_n et \mathcal{S}_n , comme le montrent les figures 14 et 15 (voir Section 9). Or, ces deux matrices engendrent le groupe symplectique

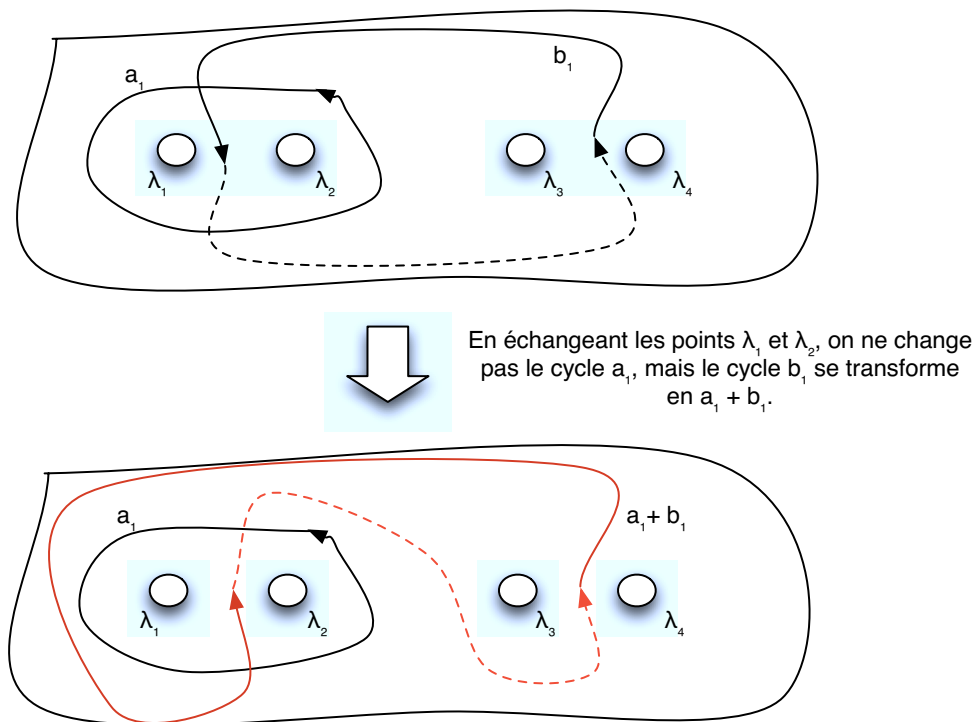
FIGURE 14. Théorème d'Arnold $n=3,4$



Considérant \mathcal{S}_n , il est possible de permuter, de manière continue, tous les points de ramification (puisque aucun de ces points ne se trouve à l'infini). Voici donc les matrices de permutation \mathcal{S}_n qui représentent l'application de la tresse sur les points de ramification λ_i et μ_i :

Ainsi, si on souhaite appliquer la tresse σ_{ij} qui permute les points de ramification λ_i et λ_j , et seulement ces deux points, il suffit de prendre

FIGURE 15. Théorème d'Arnold $n=3,4$



On veut montrer que . Il suffit donc de montrer qu'avec les cinq matrices telles qu'identifiées plus haut, on peut obtenir les matrices et de la section qui sont les matrices qui engendrent le groupe symplectique . En outre, on a :

et enfin,

ce qui achève de montrer que .

8. Échange de Points

Dans cette section, nous donnerons une explication sommaire de comment on en arrive à échanger des points et comment cela modifie-t-il les cycles sur la surface.

D'abord, il faut commencer par visualiser la surface S privée de n points identifiés, soit $S \setminus \{P_1, \dots, P_n\}$. Ensuite, on prend un autre point, Q , différent des précédents, et on considère les lacets γ_i , basés au point Q , tel que le lacet γ_i entoure le point P_i , une base de $\pi_1(S \setminus \{P_1, \dots, P_n\})$ comme le montre la figure 16 dans le cas où n est égal à 4. La composition des chemins γ_i et γ_j se fait de la manière suivante : si on souhaite calculer $\gamma_i \gamma_j$, on parcourt d'abord γ_i , puis on parcourt γ_j . On note au passage que, comme on se trouve sur S , qui est équivalent à la sphère de Riemann, telle que présentée dans la figure 2 de la section 3, alors la composition $\gamma_i \gamma_j$, qui est une boucle centrée en Q et entourant P_i et P_j est topologiquement équivalente à l'identité.

Maintenant, on veut pouvoir visualiser l'action du groupe de tresses sur les lacets que l'on vient de définir. Le groupe de tresses agissant sur n lacets, est le groupe de tresses à n générateurs, que l'on notera T_n , tel que T_n permute de manière continue les points P_i et P_j dans le sens anti-horaire, comme on peut le voir dans la figure 17, qui représente l'action de T_n . On peut voir que l'on permute, de manière continue, les points P_i et P_j , dans le sens anti-horaire.

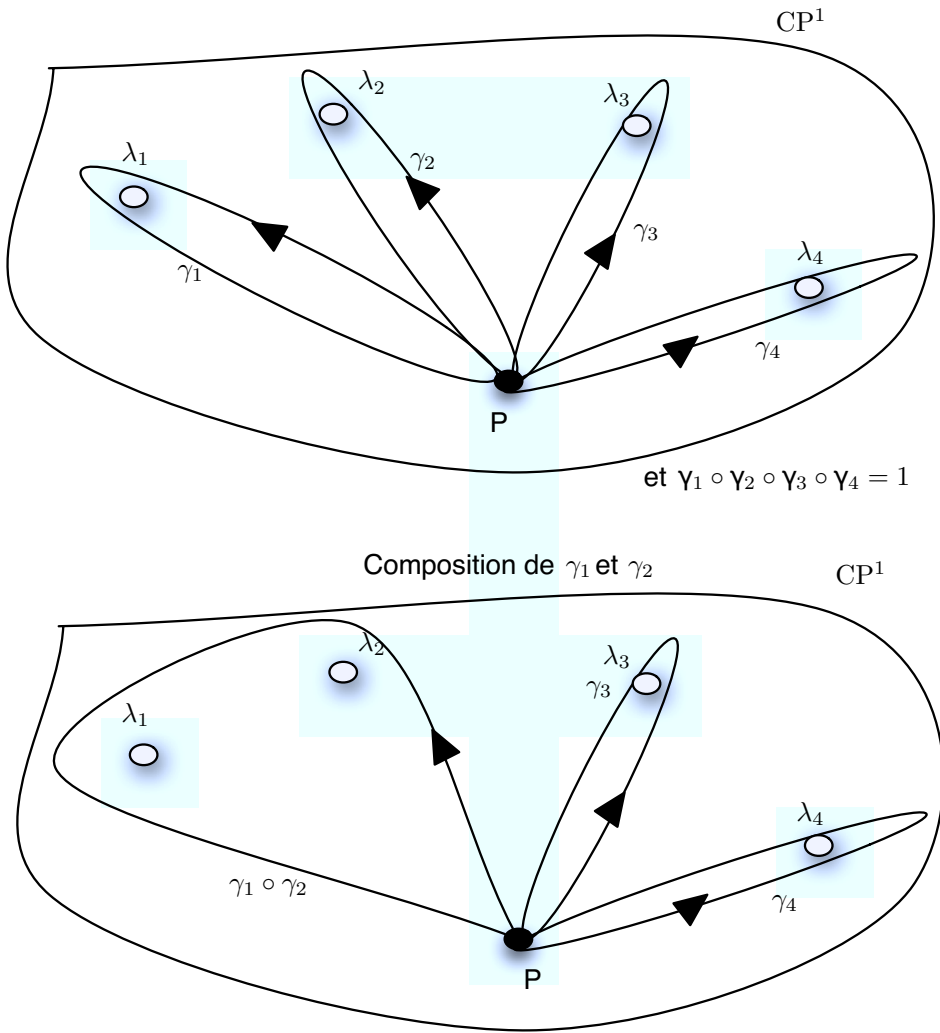
Ainsi, pour T_n , on a :

Avant de visualiser l'action du groupe de tresser sur un revêtement ramifié d'une surface hyperelliptique, rappelons-nous ce qu'était la base canonique du groupe d'homologie de cette surface. La base canonique peut être vue comme la figure 13. Maintenant, effectuons une projection des éléments de la base canonique, c'est-à-dire les cycles, sur S . La figure 18 nous montre la projection du cycle α_i et du cycle β_i sur S , passant par le point Q , et tel que les points P_i représentent la projection des points de ramification. De cette manière, on peut bien voir que la projection du cycle α_i sur S est égal à γ_i et que la projection du cycle β_i sur S est égal à γ_j .

On peut aussi faire le contraire, c'est-à-dire partir d'un cycle sur S et lui associer un cycle sur la surface hyperelliptique. Cette action s'appelle relèvement. Il y a deux relèvements pour chaque cycle : celui qui relève le point P_i sur le premier feuillet ainsi que celui qui relève le point P_j sur le second feuillet. Nous choisissons le relèvement sur le premier feuillet. Pour une surface hyperelliptique à n points de ramification, on a :

Pour visualiser l'action du groupe de tresses sur la base canonique d'homologie d'une surface hyperelliptique ayant pour points de ramification P_1, \dots, P_n , il

FIGURE 16. Lacets



faut visualiser l'action du groupe de tresses sur la surface privée de points et ensuite, on relève les cycles résultants sur la surface.
Par exemple, pour et la base canonique de la figure 13, on a le calcul suivant :

et

tel que nous l'obtenions à la figure 15. De plus,

et

tel que nous l'obtenions à la figure 14.

Références

- [1] V. I. Arnol'd, Remark on the branching of hyperelliptic integrals as functions of the parameters, Moscow State University. Translated from Funktsional'nyi Analiz i Ego Prilozheniya, Vol. 2, No. 3, pp. 1-3, July-September, 1968.

FELIX CAMIRAND, DÉPARTEMENT DE MATHÉMATIQUES, UNIVERSITÉ DE SHERBROOKE
Courriel: `Felix.Camirand.Lemyre@usherbrooke.ca`

FIGURE 17. Action de

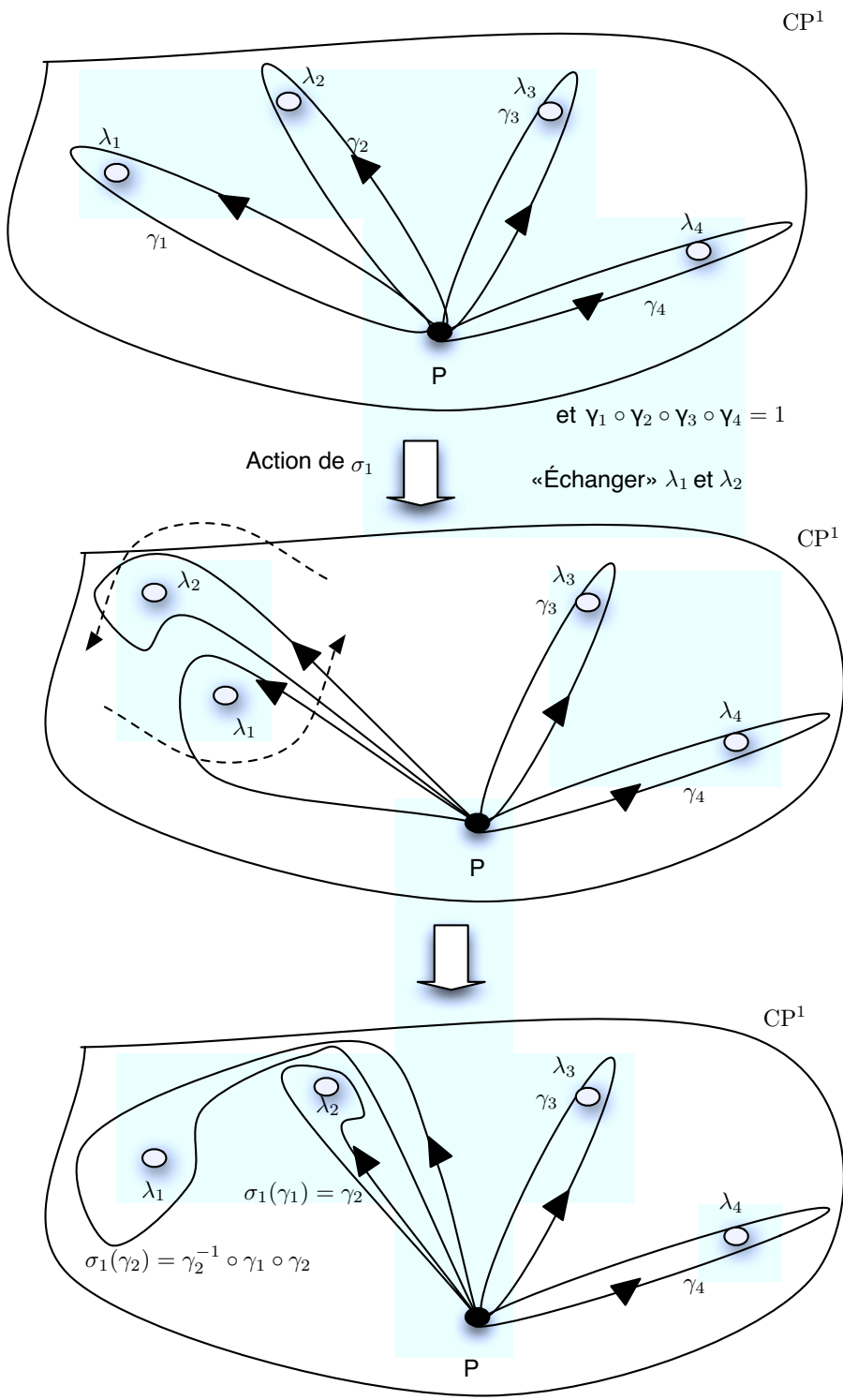
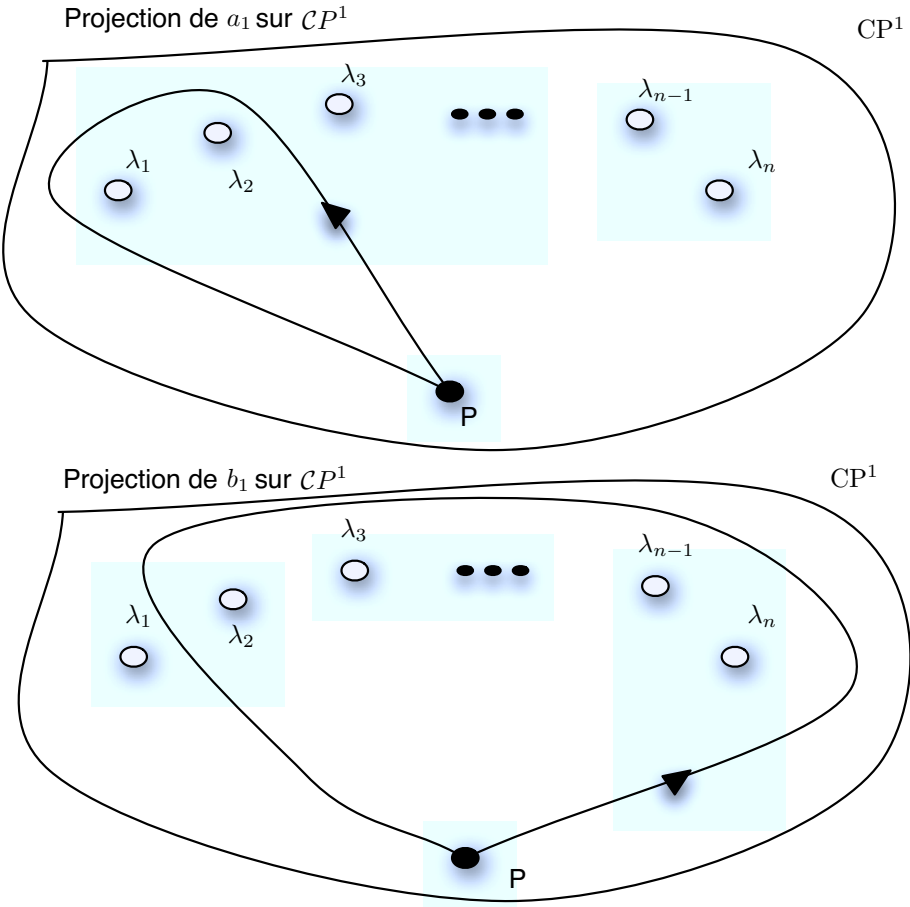


FIGURE 18. Projection sur



LES ALGÈBRES AMASSÉES: DÉFINITIONS DE BASE ET RÉSULTATS

DAVID BOULET-ST-JACQUES

1. Introduction

C'est au début des années 2000 que Sergey Fomin et Andreï Zelevinsky ont introduit la classe des algèbres amassées [FZ02]. Leur but principal était alors de fournir un cadre adéquat pour comprendre l'aspect combinatoire de la positivité totale et des bases canoniques duales. Bien que cet objectif ne soit pas encore atteint, le cadre des algèbres amassées s'est révélé extrêmement fructueux pour les mathématiques, puisqu'il a permis de relier des domaines qui, à prime abord, n'avaient pas de liens entre eux. Parmi ces domaines, on compte la géométrie de Poisson, la physique mathématique et la théorie de Lie.

L'objectif de cet article est de fournir une brève initiation à cette classe d'algèbres. La section 2 présentera les algèbres amassées sans coefficients tandis que la section 3 abordera le phénomène Laurent et la conjecture de positivité. Enfin la section 4 présentera les algèbres amassées avec coefficients.

2. Algèbres amassées sans coefficients

2.1. Définitions initiales

Afin de commencer notre étude des algèbres amassées, nous introduirons quelques notions fondamentales.

Définition 2.1.

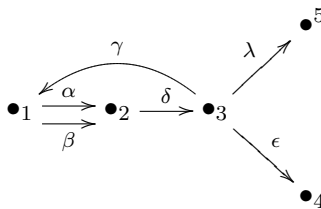
Une **\mathbb{Z} -algèbre**, ou anneau, est un quadruplet $(A, +, \cdot, 1)$ (ou plus brièvement A) formé d'un groupe abélien $(A, +)$ muni d'une opération interne $A \times A \rightarrow A$ et d'un élément $1 \in A$ satisfaisant aux axiomes suivants :

- (1) $a \cdot (b \cdot c) = (a \cdot b) \cdot c$ pour tous $a, b, c \in A$,
- (2) Il existe $1 \in A$ tel que $1 \cdot a = a \cdot 1 = a$ pour tous $a \in A$,
- (3) $a \cdot (b + c) = a \cdot b + a \cdot c$ pour tous $a, b, c \in A$,
- (4) $(a + b) \cdot c = a \cdot c + b \cdot c$ pour tous $a, b, c \in A$.

Je tiens à remercier Myriam Chabot et Guillaume Douville pour l'aide qu'ils m'ont apportée dans l'élaboration de ce travail et pour les précieuses discussions relatives à cet article. De plus, je remercie Ibrahim Assem pour les nombreux conseils et l'encadrement qu'il m'a fourni. Finalement, je remercie Ibrahim Assem et le LaCIM pour le financement de cette recherche.

Les algèbres amassées sont des \mathbb{Z} -algèbres possédant une riche structure combinatoire. Ce sont d'ailleurs des outils combinatoires qui permettent de définir les algèbres amassées.

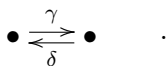
Un **carquois** est un quadruplet $Q = (Q_0, Q_1, s, b)$ où Q_0 est un ensemble dont les éléments sont appelés *sommets*, Q_1 est un ensemble dont les éléments sont appelés *flèches*, et s, b sont des fonctions de Q_1 dans Q_0 . Les fonctions $s, b : Q_1 \rightarrow Q_0$ associent à chaque $\gamma \in Q_1$ sa *source* $s(\gamma)$ et son *but* $b(\gamma)$. Soit $k \in Q_0$ un sommet du carquois. On dit que k est une **source** (ou un **puits**) de Q s'il n'existe aucune flèche $\alpha \in Q_1$ telle que $b(\alpha) = k$ (ou que $s(\alpha) = k$, respectivement). Par exemple,



est un carquois ayant 5 sommets $\{1, \dots, 5\}$ et 6 flèches $\{\alpha, \beta, \dots, \epsilon, \lambda\}$. Les sommets 4 et 5 sont des puits, mais il n'y a aucune source. Dans notre étude, nous ne considérerons que des carquois finis sans boucles ni 2-cycles. Une **boucle** est une flèche γ dans Q_1 telle que $s(\gamma) = b(\gamma)$. Une boucle est donc un carquois de la forme



Un **2-cycle** est une paire (γ, δ) de flèches dans Q_1 telle que $s(\gamma) = b(\delta)$ et $s(\delta) = b(\gamma)$. Un 2-cycle est donc un carquois de la forme



Ainsi, dans ce texte, le mot carquois désignera toujours un carquois fini sans boucles ni 2-cycles. Les algèbres amassées sont définies à l'intérieur du corps ambiant des fractions rationnelles.

Définition 2.2.

Soit $\mathbb{Z}[x_1, \dots, x_n]$ l'anneau des polynômes à coefficients entiers en n indéterminées x_1, \dots, x_n . Le **corps des fractions rationnelles** $\mathbb{Q}(x_1, \dots, x_n)$ se définit de la façon suivante

$$\mathcal{F} = \mathbb{Q}(x_1, \dots, x_n) = \left\{ \frac{f(x_1, \dots, x_n)}{g(x_1, \dots, x_n)} \mid f, g \in \mathbb{Z}[x_1, \dots, x_n] \right\},$$

muni des opérations usuelles des fractions. Par la suite, ce corps sera appelé **corps ambiant**.

Définition 2.3.

On appelle **amas initial** l'ensemble fini $X = \{x_1, \dots, x_n\}$ des n générateurs du corps ambiant $\mathcal{F} = \mathbb{Q}(x_1, \dots, x_n)$. Ces variables sont appelées **variables initiales**.

Nous continuons notre étude des algèbres amassées, en définissant la notion de mutation.

2.2. Mutations

Définition 2.4.

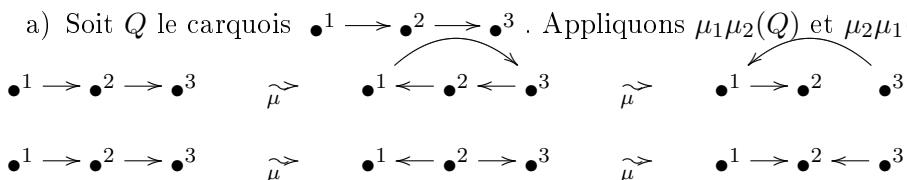
La **mutation** d'un carquois Q en un sommet k transforme Q en un nouveau carquois $Q' = \mu_k(Q)$ obtenu au moyen des étapes suivantes :

- (1) Pour tout chemin de longueur 2 de type $i \rightarrow k \rightarrow j$, on ajoute une flèche $i \rightarrow j$.
- (2) On renverse le sens de toutes les flèches incidentes à k .
- (3) On élimine un à un tous les 2-cycles.

On définit alors une **suite de mutations** comme étant la composition de plusieurs mutations. Considérons un carquois Q ayant plus de 6 sommets, la composition de mutations $\mu_3\mu_2\mu_6(Q)$ indique que l'on doit appliquer une mutation au sommet 6, puis au sommet 2 et finalement au sommet 3 du carquois Q .

Exemple 2.5.

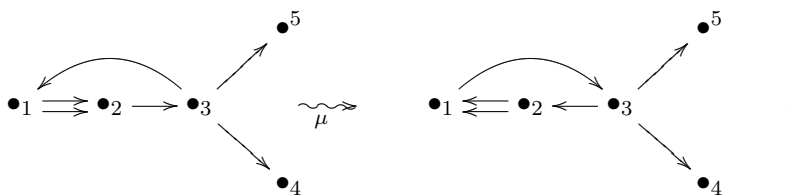
a) Soit Q le carquois $\bullet^1 \rightarrow \bullet^2 \rightarrow \bullet^3$. Appliquons $\mu_1\mu_2(Q)$ et $\mu_2\mu_1(Q)$.



On voit que les mutations d'un carquois Q ne commutent pas

$$\mu_1\mu_2(Q) \neq \mu_2\mu_1(Q) .$$

b)



Il suit de la définition que la mutation est un processus local. De plus, il serait intéressant de savoir ce qui arriverait si on appliquait une mutation plusieurs fois consécutives à un même sommet.

Lemme 2.6. *Pour tout carquois Q , on a $\mu_k^2(Q) = Q$. En d'autres termes, la mutation est involutive.*

DÉMONSTRATION. Soit Q un carquois. Notons que la mutation est un processus local. Ainsi, si $\mu_k^2 = \mathbb{I}$ pour toutes les flèches incidentes à k et tous les sommets voisins de k , ce qu'on appelle le voisinage de k , alors on peut conclure que $\mu_k^2(Q) = Q$. Vérifions donc que $\mu_k^2(Q) = Q$ pour le voisinage de k . En appliquant la définition 2.4, on obtient les étapes suivantes :

- (1) On crée une flèche $i \rightarrow j$ pour chaque chemin de longueur 2 de la forme $i \rightarrow k \rightarrow j$. Soit n le nombre de flèches $i \rightarrow j$ ainsi obtenues, alors $n \geq 0$.
- (2) On inverse toutes les flèches incidentes à k .
- (3) On enlève tous les 2-cycles créés par l'étape 1. Soit m le nombre de flèches $j \rightarrow i$. Si $0 \leq m \leq n$, il reste alors $n - m$ flèches $i \rightarrow j$. Si $m > n$, il reste $m - n$ flèches $j \rightarrow i$.
- (4) La deuxième mutation crée une flèche $i' \rightarrow j'$ pour chaque paire de flèches $i' \rightarrow k \rightarrow j'$. Or, les flèches ainsi créées seront des flèches $j \rightarrow i$, ceci étant attribuable à l'étape 2, et il y en aura exactement n .
- (5) On inverse toutes les flèches incidentes à k . Or, puisqu'on les avait déjà inversées à l'étape 2, l'étape 5 redonne exactement les mêmes flèches que dans Q .
- (6) Il ne reste plus qu'à enlever les 2-cycles créés à l'étape 4. Si on avait $m > n$, on a vu à l'étape 3 qu'il resterait $m - n$ flèches $j \rightarrow i$. Or, on vient de créer n flèches $j \rightarrow i$ à l'étape 4. Il y a donc $m - n + n = m$ flèches $j \rightarrow i$, soit le même nombre que dans Q . Si $m \leq n$, on a vu qu'il resterait $n - m$ flèches $i \rightarrow j$. Or, comme on vient de créer n flèches $j \rightarrow i$, il nous resterait $n - (n - m) = m$ flèches $j \rightarrow i$ après la suppression des 2-cycles. Ce nombre est le même que dans Q . Ainsi, $\mu_k^2(Q) = Q$.

La mutation de carquois permet de transformer un carquois Q en un nouveau carquois Q' . De même, la mutation d'amas permet de transformer un amas X en un autre amas X' en ne changeant qu'une variable à la fois. On appelle **amas** un ensemble de n générateurs algébriquement indépendants du corps ambiant \mathcal{F} , obtenu par une suite de mutations (au sens ci-bas) à partir de l'amas initial.

Définition 2.7.

Considérons l'amas $X = \{x_1, x_2, \dots, x_n\}$. La **mutation** μ_k d'un amas X en une variable x_k transforme X en un nouvel amas $X' = \mu_k(X) = (X \setminus \{x_k\}) \cup \{x_k\}$, où

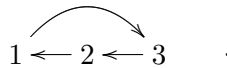
$$x_k = \frac{1}{x_k} \left(\prod_{\{\alpha | b(\alpha)=k\}} x_{b(\alpha)} + \prod_{\{\beta | s(\beta)=k\}} x_{s(\beta)} \right).$$

On convient que le produit vide vaut 1. Notons que la nouvelle variable x'_k appartient au corps ambiant \mathcal{F} . Muter un amas revient donc à transformer une des variables de cet amas. De plus, selon la définition, cette transformation dépend de l'orientation du carquois sur lequel on mute et, comme pour les carquois, il s'agit d'une opération locale.

Exemple 2.8.

- a) Soit Q le carquois $1 \longrightarrow 2 \longrightarrow 3$ et $\{x_1, x_2, x_3\}$ l'amas initial. Appliquant μ_2 à cet amas, on a que x_1 et x_3 sont inchangées alors que x_2

devient $x'_2 = \frac{x_1 + x_3}{x_2}$. Le carquois est devenu



- b) Soit Q' le carquois 1 à un seul sommet et sans flèches, et l'amas initial $X' = \{x_1\}$. Appliquant μ_1 à cet amas, on a que x_1 devient $x'_1 = \frac{2}{x_1}$. Le carquois est inchangé par cette mutation. En appliquant μ_1 à notre nouvel amas, on obtient que x'_1 devient $x''_1 = x_1$.

Lemme 2.9. *La mutation de variables est involutive.*

DÉMONSTRATION. Soit $X = \{x_1, \dots, x_n\}$ un amas. En appliquant $\mu_k(X)$, on obtient $X' = \{x_1, \dots, x'_k, \dots, x_n\}$. De plus, en appliquant $\mu_k(X')$ on obtient $X'' = \{x_1, \dots, x''_k, \dots, x_n\}$. Ainsi, il suffit de démontrer que $x''_k = x_k$ pour obtenir que $X'' = X$ et donc que la mutation de variables est involutive. Or, en appliquant la définition, on obtient

$$\begin{aligned} x''_k &= \frac{1}{x'_k} \left(\prod_{\{\alpha | b(\alpha)=k\}} x_{b(\alpha)} + \prod_{\{\beta | s(\beta)=k\}} x_{s(\beta)} \right) \\ &= \frac{x_k}{\prod_{\{\alpha | b(\alpha)=k\}} x_{b(\alpha)} + \prod_{\{\beta | s(\beta)=k\}} x_{s(\beta)}} \left(\prod_{\{\alpha | b(\alpha)=k\}} x_{b(\alpha)} + \prod_{\{\beta | s(\beta)=k\}} x_{s(\beta)} \right) \\ &= x_k \quad . \end{aligned}$$

2.3. Algèbres amassées sans coefficients

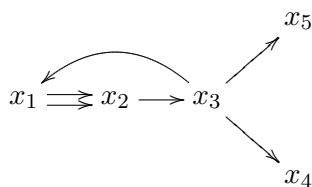
Les algèbres amassées sans coefficients sont définies à partir d'un carquois initial et d'un amas initial auxquels on applique des mutations.

Définition 2.10.

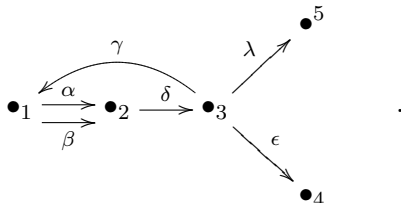
On appelle **graine initiale** une paire (X, Q) où $X = \{x_1, \dots, x_n\}$ est un amas dit initial et Q est un carquois. Une **graine** G dans \mathcal{F} est une paire (X', Q') , où Q' est le carquois obtenu de Q par une suite de mutations du carquois initial Q et $X' = \{x'_1, \dots, x'_n\}$ est l'ensemble de n variables obtenues de l'amas initial X par la même suite de mutations. On voit que X' est un amas tel que défini précédemment.

Afin de représenter une graine (X, Q) , on associe au sommet i de Q la variable x_i de X . Ainsi, on obtient un carquois dont l'ensemble des sommets Q_0 est indexé

par les variables de l'amas X . Ainsi,



est une graine associée au carquois



Définition 2.11.

Soit l'amas initial $X = \{x_1, x_2, \dots, x_n\}$ et soit $X_\alpha = \mu_{1_\alpha} \mu_{2_\alpha} \dots \mu_{t_\alpha}(X)$ l'amas obtenu par la suite de mutations $\mu_{1_\alpha} \mu_{2_\alpha} \dots \mu_{t_\alpha}$. Alors $\bigcup_{\alpha} X_\alpha$ est l'ensemble de toutes les variables amassées. On définit alors l'algèbre amassée $(X, Q) = \mathbb{Z}[\bigcup_{\alpha} X_\alpha]$ comme étant l'anneau des polynômes à coefficients entiers en les variables amassées, muni des opérations usuelles des polynômes et du produit usuel par un entier.

En résumé, on considère l'ensemble de toutes les variables amassées obtenues par suite de mutations comme ensemble de générateurs de notre algèbre. De cet ensemble de variables, on obtient tous les polynômes possibles à coefficients entiers. Chaque polynôme est alors un élément de l'algèbre amassée. Chaque algèbre amassée dépend uniquement de la graine initiale.

Exemple 2.12.

$$\begin{array}{ccccc}
 x_1 \longrightarrow x_2 & \xrightarrow{\mu} & \frac{1+x}{x} \longleftarrow x_2 & \xrightarrow{\mu} & \frac{1+x}{x} \longrightarrow \frac{1+x+x}{x \ x} \\
 \mu \downarrow & & & & \mu \downarrow \\
 x_1 \longleftarrow \frac{1+x}{x} & = & \frac{1+x}{x} \longrightarrow x_1 & \xleftarrow{\mu} & \frac{1+x}{x} \longleftarrow \frac{1+x+x}{x \ x}
 \end{array}$$

Puisque la mutation est involutive nous avons ici tous les cas possibles de suites de mutations. Ainsi, on obtient l'algèbre amassée suivante :

$$= (\{x_1, x_2\}, 1 \rightarrow 2) = \mathbb{Z} \left[x_1, x_2, \frac{1+x_2}{x_1}, \frac{1+x_1}{x_2}, \frac{1+x_1+x_2}{x_1 x_2} \right]$$

Un élément de pourrait alors être, par exemple :

$$18 \left(\left(\frac{1+x_2}{x_1} \right)^6 + \left(\frac{1+x_1+x_2}{x_1 x_2} \right)^{17} \right) .$$

2.4. Mutations de matrices

Avant de parler du lien entre carquois et matrices, rappelons que la **matrice transposée** d'une matrice carrée $B_n = [b_{ij}]_n$ d'ordre n sur un anneau K est la matrice

$$B_n^T = [b'_{ij}]_n, \text{ où } b'_{ij} = b_{ji}.$$

Transposer une matrice revient à échanger la i^e ligne avec la j^e colonne. De plus, une matrice **antisymétrique** est une matrice B_n carrée à coefficients dans K telle que

$$B_n^T = -B_n,$$

ce qui équivaut, si $B_n = [b_{ij}]$, à dire que $b_{ii} = 0$ pour tout i et $b_{ij} = -b_{ji}$ pour $i \neq j$. Il est possible d'associer un carquois à une matrice anti-symétrique. Le nombre de lignes (ou de colonnes) est égal au nombre de sommets du carquois. Le $(i, j)^{\text{ème}}$ coefficient de la matrice est le nombre de flèches allant du sommet i au sommet j . S'il y a des flèches allant plutôt du sommet j au sommet i , on prend encore le nombre de flèches de j vers i , mais précédé d'un signe négatif.

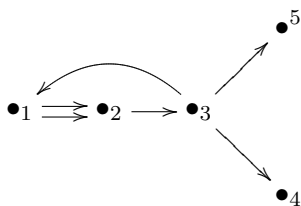
Proposition 2.13. *Il existe une bijection ϕ entre l'ensemble des carquois (sans boucles ni 2-cycles) et celui des matrices antisymétriques à coefficients entiers. Celle-ci est donnée par*

$$\phi: Q \mapsto B_n = [b_{ij}]_n$$

où b_{ij} est le nombre de flèches allant de i vers j dans Q .

DÉMONSTRATION. Soit Q un carquois. On définit une fonction f de l'ensemble des carquois sans boucles ni 2-cycles vers l'ensemble des matrices antisymétriques de dimensions $n \times n$, où $n = |Q_0|$, par $f(Q) = [b_{ij}]$, où b_{ij} est le nombre de flèches allant de i vers j dans Q , et une fonction g de l'ensemble des matrices antisymétriques de dimensions $n \times n$ vers l'ensemble des carquois sans boucles ni 2-cycles par $g([b'_{ij}]) = Q'$, où $[b'_{ij}]$ est une matrice antisymétrique et Q' est le carquois obtenu en plaçant b'_{ij} flèches allant du sommet i au sommet j . En appliquant $f(Q)$, on obtient une matrice B dont la diagonale est constituée de 0 puisque le carquois Q est sans boucles. De plus, en se référant à la définition de f , on constate que la matrice sera antisymétrique. Il est donc possible d'appliquer $g(B)$. Or, $g(B)$ nous donne un carquois Q' qui sera sans boucles ni 2-cycles, puisque la diagonale principale de B est nulle et que la matrice est antisymétrique. Finalement, en vertu de la définition des fonctions f et g , le nombre de flèches allant de i à j sera le même dans Q' que dans Q . Il est alors facile de vérifier que $g(f(Q)) = Q$ et $f(g([b_{ij}])) = [b_{ij}]$.

Exemple 2.14. Au moyen de la bijection de la proposition 2.13, le carquois



s'applique sur la matrice $B \in M_{5 \times 5}(\mathbb{Z})$,

$$\text{où } B = \begin{pmatrix} 0 & 2 & -1 & 0 & 0 \\ -2 & 0 & 1 & 0 & 0 \\ 1 & -1 & 0 & 1 & 1 \\ 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{pmatrix}.$$

Comme il est possible de muter les carquois, il est aussi possible de muter les matrices antisymétriques.

Proposition 2.15.

a) Soit $B_n = [b_{ij}]_n$ une matrice associée à un carquois. On a alors que les éléments de la matrice obtenue par une mutation en k sont

$$b'_{ij} = \begin{cases} -b_{ij} & \text{si } k = i \text{ et } k = j \\ b_{ij} + \frac{|b_{ik}|b_{kj} + b_{ik}|b_{kj}|}{2} & \text{sinon.} \end{cases}$$

b) Soit $X = \{x_1, x_2, \dots, x_n\}$ l'amas initial. Alors on a que $\mu_k(X) = X' = (X \setminus \{x_k\}) \cup \{x'_k\}$, où

$$x'_k = \frac{\prod_{b_{ik} > 0} x_i^{b_{ik}} + \prod_{b_{ik} < 0} x_i^{-b_{ik}}}{x_k}.$$

DÉMONSTRATION. Il s'agit de la traduction en termes de matrices de la mutation de carquois.

On sait que les algèbres amassées sont définies par un ensemble de générateurs. Or, tel que discuté précédemment, cet ensemble peut être soit fini, soit infini. Fomin et Zelevinsky ont classifié les algèbres amassées dont l'ensemble de générateurs est fini [FZ03].

2.5. Classification des algèbres amassées de type fini

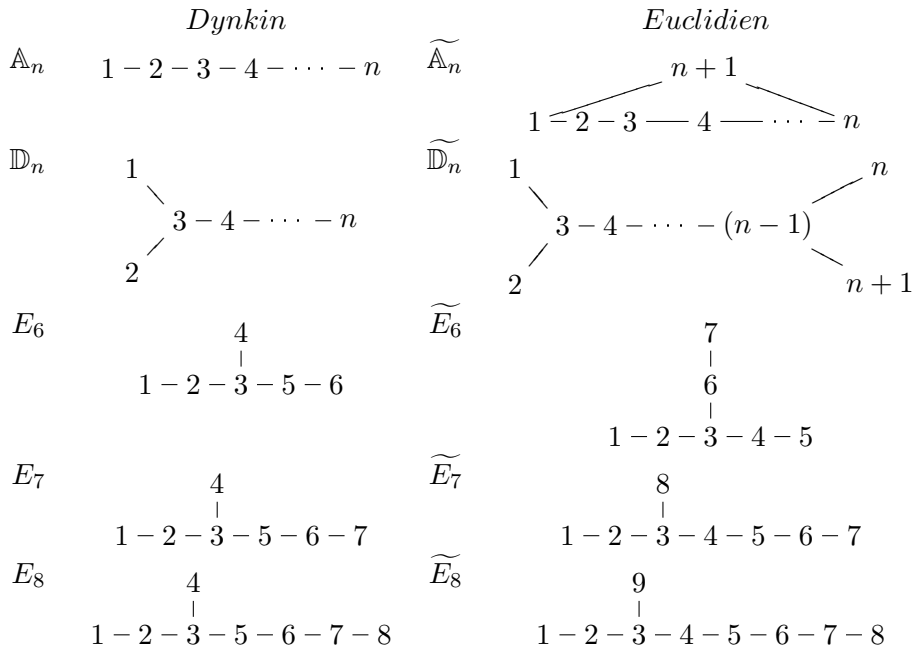
Définition 2.16.

Une algèbre amassée est dite **de type fini** si l'ensemble de ses variables amassées est fini.

Comme les algèbres amassées dépendent uniquement du choix de la graine initiale, il est raisonnable de penser que le choix du carquois Q de notre graine initiale pourrait influencer sur le fait que l'algèbre amassée soit de type fini ou pas.

Définition 2.17.

Les **carquois de type Dynkin et Euclidien** sont les carquois dont le graphe sous-jacent est de l'un des types suivants :



Théorème 2.18. Une algèbre amassée (X, Q) est de type fini si et seulement si une de ses graines a le carquois dont le graphe sous-jacent est un diagramme de Dynkin.

DÉMONSTRATION. Pour une preuve de ce théorème, on réfère le lecteur à [FZ03].

3. Résultats importants

Cette section présentera notre résultat principal sur les algèbres amassées, ainsi qu'une conjecture sur les variables amassées

3.1. Phénomène Laurent

L'un des aspects les plus surprenants de la théorie des algèbres amassées est ce qu'on appelle le phénomène Laurent.

Définition 3.1.

Soit $\{x_1, x_2, \dots, x_n\}$ un ensemble de variables.

L'ensemble des **polynômes de Laurent** en ces variables est l'anneau

$$\mathbb{Z} [x_1, x_2, \dots, x_n, x_1^{-1}, x_2^{-1}, \dots, x_n^{-1}] .$$

Il s'agit de toutes les expressions de la forme

$$p = \sum_{k \in \mathbb{Z}} a_k x_1^{m_{1,k}} x_2^{m_{2,k}} \cdots x_n^{m_{n,k}} ,$$

où les $m_{i,k}$ sont des entiers et où les $a_k \neq 0$ pour un nombre fini de k entiers. Puisque la somme et le produit de polynômes de Laurent sont des polynômes de Laurent, cet ensemble est un sous-anneau du corps ambiant $\mathcal{F} = \mathbb{Q}(x_1, \dots, x_n)$.

Notons qu'en ramenant à un dénominateur commun, on voit que tout polynôme de Laurent s'écrit sous la forme

$$p = \frac{\sum_{k \in \mathbb{Z}} b_k x_1^{p_k} x_2^{p_k} \dots x_n^{p_k}}{x_1^{s_1} x_2^{s_2} \dots x_n^{s_n}}$$

avec les p_{ik} et les s_i des entiers positifs et les $b_k \neq 0$ seulement pour un nombre fini de k .

Exemple 3.2. Soit $X = \{x_1\}$. Alors

$$p = \frac{1}{x_1}$$

est un polynôme de Laurent n'appartenant pas à l'algèbre amassée de type \mathbb{A}_1 . En effet, si on veut que ce polynôme soit dans l'algèbre amassée, il devrait être égal à une expression de la forme

$$\sum_{n,m} a_{n,m} x_1^n \left(\frac{2}{x_1}\right)^m = \sum_{n,m} a_{n,m} 2^m x_1^{n-m},$$

avec $n, m \geq 0$. Or, si cette expression égale p , on doit avoir $n - m = -1$. Mais alors, si $n - m = -1$, on obtient que $m \geq 1$ ce qui contredit l'égalité entre le numérateur de cette expression et celui de p . Cependant,

$$\begin{aligned} p' &= 3 \left(\frac{2}{x_1}\right)^2 + 4x_1^{13} \\ &= \frac{12 + 4x_1^{15}}{x_1^2} \end{aligned}$$

est un polynôme de Laurent appartenant à l'algèbre amassée de type \mathbb{A}_1 .

Grâce à la définition 3.1, il est possible d'énoncer un théorème central des algèbres amassées. Ce théorème fut l'un des premiers prouvés par Fomin et Zelevinsky [FZ02] et a grandement contribué à l'intérêt que l'on porte à cette classe d'algèbres.

Théorème 3.3. *Soit $(Y = \{y_1, y_2, \dots, y_n\}, Q_Y)$ une graine quelconque d'une algèbre amassée. Alors toute variable amassée s'écrit comme polynôme de Laurent en les variables y_1, y_2, \dots, y_n . En particulier,*

$$\subseteq \mathbb{Z} [y_1, \dots, y_n, y_1^{-1}, \dots, y_n^{-1}].$$

DÉMONSTRATION. Pour une preuve de ce théorème, on réfère le lecteur à l'article [FZ02] de Fomin et Zelevinsky.

3.2. Conjecture de positivité

Soient $p(x) = \sum_i p_i x^i$, $q(x) = \sum_i q_i x^i$ et $r(x) = \sum_i r_i x^i$, des polynômes à coefficients dans \mathbb{Z} tels que

$$\frac{p(x)}{q(x)} = r(x).$$

Il est facile de voir que même si tous les p_i et tous les q_i sont non-négatifs, alors les r_i peuvent être négatifs. En effet, on pourrait considérer le cas où $p(x) = x^3 + 1$ et $q(x) = x + 1$. On aurait alors que $r(x) = x^2 - x + 1$.

Les variables amassées sont des polynômes de Laurent, et les exemples précédents montrent que les coefficients de ces polynômes sont des entiers positifs. Il serait utile de savoir si c'est toujours le cas et pourquoi. Ce problème est encore ouvert dans certains cas et fait présentement l'objet d'une conjecture.

Conjecture de positivité. *Soit $(Y = \{y_1, y_2, \dots, y_n\}, Q_Y)$ une graine quelconque d'une algèbre amassée. Alors toute variable amassée est dans $\mathbb{Z}_{\geq 0} [y_1, \dots, y_n, y_1^{-1}, \dots, y_n^{-1}]$. C'est-à-dire qu'elle s'écrit sous la forme*

$$\frac{\sum_{k \in \mathbb{Z}} b_k y_1^{p_k} \dots y_n^{p_{nk}}}{y_1^{s_1} \dots y_n^{s_n}}, \text{ avec } b_k \geq 0.$$

La conjecture a été démontrée par Musiker, Schiffler et Williams en 2009, [MSW09], pour toute algèbre amassée provenant d'une surface marquée. Cela inclut les algèbres amassées des types :

$$\mathbb{A}_n, \quad \mathbb{D}_n, \quad \tilde{\mathbb{A}}_n, \quad \tilde{\mathbb{D}}_n.$$

On réfère à [Dou11] pour de l'information sur les surfaces marquées.

4. Algèbres amassées avec coefficients

Les deux premières sections ont présenté des propriétés des algèbres amassées sans coefficients. Or, ces algèbres sont un cas particulier des algèbres amassées avec coefficients. Dans cette dernière section, nous présentons cette classe plus large d'algèbres amassées.

4.1. Définitions initiales

Puisque les algèbres amassées avec coefficients généralisent celles sans coefficients, nous ne donnerons que les définitions différentes ou nouvelles par rapport à celles vues en 2.1. De plus, il faudra généraliser la notion de mutation.

Notation 4.1. *Soit Q un carquois et $i, j \in Q_0$. On note $Q_1(i, j)$ l'ensemble des flèches de Q_1 allant de i vers j .*

Définition 4.2.

Un **carquois gelé** est une paire (Q, F) , où Q est un carquois et F est un sous-ensemble de Q_0 tel que pour tous i, j dans F , il n'existe aucune flèche entre i et j dans Q_1 . Les éléments de F sont appelés **sommets gelés**.

En posant $F = \emptyset$, on voit que la paire $(Q, F) = (Q, \emptyset)$ est en fait un carquois tel que défini à la section 2.1.

Définition 4.3.

Soit Q un carquois. On note $\hat{Q} = (\hat{Q}_0, \hat{Q}_1, s, b)$ le carquois défini par :

$$\begin{aligned}\hat{Q}_0 &= Q_o \cup Q'_o \text{ où } Q'_o = \{i' | i \in Q_0\} \\ \hat{Q}_1(i, j) &= \begin{cases} Q_1(i, j) & \text{si } i, j \in Q_0 \\ \{j' \rightarrow j\} & \text{si } j \in Q_0 \\ \emptyset & \text{sinon.} \end{cases}\end{aligned}$$

Le carquois gelé (\hat{Q}, Q'_0) , est appelé **l'extension principale** de Q .

Dans la suite de cette section, un sommet encadré représentera un sommet gelé.

Exemple 4.4. Soit $Q = 1 \longrightarrow 2 \longrightarrow 3$. Alors l'extension principale de Q est

$$(\hat{Q}, Q'_0) = \begin{array}{ccccc} & 1 & \longrightarrow & 2 & \longrightarrow & 3 \\ & \uparrow & & \uparrow & & \uparrow \\ & \boxed{1'} & & \boxed{2'} & & \boxed{3'} \end{array}$$

4.2. Mutation de carquois gelés

Comme pour les algèbres amassées sans coefficients, il faut se doter d'une notion de mutation permettant de générer un ensemble de générateurs pour nos algèbres amassées avec coefficients. Cette notion sera une légère modification de celle définie en 2.4.

Définition 4.5.

Soit (Q, F) un carquois gelé et i un sommet non-gelé. La **mutation** $\mu_i(Q, F)$ **dans la direction** i de (Q, F) transforme le carquois (Q, F) en un nouveau carquois gelé $\mu_i(Q, F) = (Q', F)$ obtenu par :

- (1) On mute en i selon la mutation usuelle ;
- (2) On retire toute flèche créée entre 2 sommets gelés.

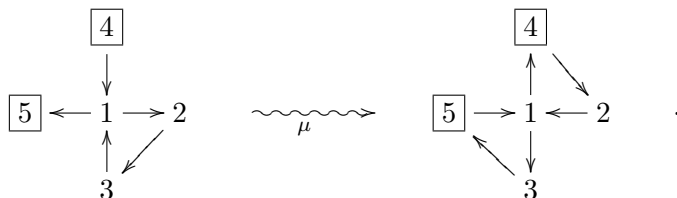
Bref, muter un carquois gelé revient à muter un carquois standard avec la restriction qu'il ne doit pas y avoir de flèches entre les sommets gelés. On peut à nouveau remarquer que si F est l'ensemble vide, alors la mutation de carquois gelés est la même que la mutation usuelle.

Lemme 4.6. Soit (Q, F) un carquois gelé. La mutation de carquois gelés est une *involution*, c'est-à-dire que $\mu_k^2(Q, F) = (Q, F)$.

DÉMONSTRATION. La preuve du lemme 2.6 s'applique avec les modifications évidentes.

Voici un exemple de mutation appliquée à un carquois gelé.

Exemple 4.7.



Remarque. On définit le type Dynkin ou Euclidien d'un carquois gelé comme étant le type Dynkin ou Euclidien de sa partie non-gelée. Ainsi, le carquois de l'exemple précédent était de type \mathbb{A}_3 .

4.3. Matrices et carquois gelés

De manière analogue à ce qui a été fait pour les carquois, il est possible d'associer une matrice à un carquois gelé.

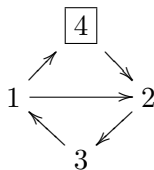
Proposition 4.8. *Il existe une bijection f entre l'ensemble des carquois gelés à $n+m$ sommets dont m sont gelés et celui des matrices de dimension $(n+m) \times m$ à coefficients entiers, celle-ci étant donnée par*

$$f : (Q, F) \longmapsto \tilde{B}_{(Q, F)}$$

où $\tilde{B}_{(Q, F)}(i, j) = |Q_1(i, j)| - |Q_1(j, i)|$.

DÉMONSTRATION. La preuve de la proposition 2.13 s'applique avec les modifications évidentes.

Exemple 4.9. Le carquois



est associé à la matrice

$$\tilde{B} = \begin{bmatrix} 0 & 1 & -1 \\ -1 & 0 & 1 \\ -1 & 1 & 0 \\ -1 & 1 & 0 \end{bmatrix}.$$

Remarque. Si $F = \emptyset$, alors la matrice associée à (Q, F) est une matrice anti-symétrique.

4.4. Graines avec coefficients et leurs mutations

Les algèbres amassées avec coefficients sont définies sur le même corps ambiant que celles sans coefficients. On rappelle au lecteur que le corps utilisé est celui des fonctions rationnelles sur l'ensemble des variables initiales. De plus, si on considère un carquois gelé (Q, F) , on convient que si $Q_0 = \{1, \dots, n, n+1, \dots, n+m\}$, alors $F = \{n+1, \dots, n+m\}$.

Définition 4.10.

Une **graine avec coefficients** est un triplet $\Sigma = (X, Y, (Q, F))$, où :

- (Q, F) est un carquois gelé ;
- $X = \{x_1, \dots, x_n\}$ est un n -uplet de variables associées aux sommets non-gelés de Q ;
- $Y = \{y_1, \dots, y_m\}$ est un m -uplet de variables associées aux sommets gelés de F .

X est appelé **l'amas de la graine** Σ et les éléments de X sont les variables non-gelées tandis que Y est appelé **l'ensemble de variables gelées de Σ** .

Soit $\Sigma = ((x_1), (x_2, x_3), \boxed{2} \longrightarrow 1 \longleftarrow \boxed{3})$ une graine avec coefficients. On la représente comme suit :

$$\boxed{x_2} \longrightarrow x_1 \longleftarrow \boxed{x_3}$$

Comme pour les graines sans coefficients, il est possible de muter une graine avec coefficients. La définition est semblable.

Définition 4.11.

Soit $\Sigma = (X, Y, (Q, F))$. Alors, pour tout i dans $Q_0 \setminus F$, la mutation de Σ en i , est $\mu_i(\Sigma) = (X', Y, \mu_i(Q, F))$, où $X' = (X \setminus \{x_i\}) \cup \{x'_i\}$ avec

$$x_i x'_i = \prod_{\{\alpha | s(\alpha)=k\}} x_{b(\alpha)} + \prod_{\{\beta | b(\beta)=k\}} x_{s(\beta)}.$$

Reprenons la graine de l'exemple précédant pour lui appliquer une mutation. Soit $\Sigma = ((x_1), (x_2, x_3), \boxed{2} \longrightarrow 1 \longleftarrow \boxed{3})$. La mutation de Σ en 1 donne alors

$$\mu_1(\Sigma) = ((x'_1), (x_2, x_3), \boxed{2} \longleftarrow 1 \longrightarrow \boxed{3})$$

où :

$$x'_1 = \frac{1 + x_2 x_3}{x_1}.$$

4.5. Algèbres amassées avec coefficients

Par analogie avec les algèbres amassées sans coefficients, on définit l'algèbre amassée associée à une graine avec coefficients Σ . Comme à la définition 2.11, on note $\bigcup_{\alpha} X_{\alpha}$ l'ensemble de toutes les variables amassées. Notons que les algèbres amassées avec coefficients sont aussi définies sur le corps ambiant $\mathcal{F} = \mathbb{Q}(x_1, \dots, x_{n+m})$.

Définition 4.12.

L'algèbre amassée avec coefficients associée à $\Sigma = (X, Y, (Q, F))$, notée Σ est :

$$\Sigma = \mathbb{Z}[y^{\pm 1} \mid y \in Y][\bigcup X_{\alpha}].$$

Bref, il s'agit de toutes les fractions rationnelles à coefficients entiers formées de variables amassées et de variables gelées.

Exemple 4.13. Soit $\Sigma = ((x_1), (x_2, x_3), \boxed{2} \longrightarrow 1 \longleftarrow \boxed{3})$.

$$\boxed{x_2} \longrightarrow x_1 \longleftarrow \boxed{x_3} \quad \rightsquigarrow_{\mu} \quad \boxed{x_2} \longleftarrow \frac{1 + x_2 x_3}{x_1} \longrightarrow \boxed{x_3}.$$

Comme la mutation est involutive, et qu'on ne peut pas muter sur les sommets gelés, on a obtenu toutes les variables amassées associées à Σ . Ainsi,

$$\begin{aligned} \bigcup X_\alpha &= \left\{ x_1, \frac{1+x_2x_3}{x_1} \right\} \quad \text{et} \quad \Sigma = \mathbb{Z} [x_2^{\pm 1}, x_3^{\pm 1}] \left[x_1, \frac{1+x_2x_3}{x_1} \right] \\ &= \mathbb{Z} \left[x_1, x_2, x_3, \frac{1}{x_2}, \frac{1}{x_3}, \frac{1+x_2x_3}{x_1} \right]. \end{aligned}$$

4.6. Changement de coefficients

Depuis le début de cette section, on voit que les algèbres amassées sans coefficients ressemblent à celles avec coefficients. Or, il existe un morphisme permettant de passer d'un type d'algèbre amassée à l'autre. Ce changement fera l'objet de cette dernière section.

Notation 4.14. Soit (Q, F) un carquois gelé. On note U le carquois obtenu de Q en supprimant les sommets gelés.

On cherche maintenant à comparer (Q, F) avec U . Nous aurons besoin au préalable de la notion de morphisme d'anneaux.

Définition 4.15.

Soient A et B des anneaux. Soient $+$, \cdot et 1 respectivement l'opération additive, multiplicative et le neutre multiplicatif des anneaux, **un morphisme d'anneaux** $f : A \rightarrow B$ est une application telle que :

- Pour tous a, b dans A , on a $f(a + b) = f(a) + f(b)$;
- Pour tous a, b dans A , on a $f(a \cdot b) = f(a) \cdot f(b)$ et ;
- $f(1) = 1$.

Grâce à la définition de morphisme d'anneau, on peut définir un morphisme entre les corps ambiants de nos algèbres amassées.

Définition 4.16.

Le **morphisme de spécialisation** ρ est le morphisme d'anneaux défini comme suit

$$\begin{aligned} \rho : \mathbb{Q}(x_1, \dots, x_{m+n}) &\longrightarrow \mathbb{Q}(x_1, \dots, x_n) \\ x_i &\longmapsto \begin{cases} x_i & \text{si } 1 \leq i \leq n \\ 1 & \text{si } n < i \leq n+m \end{cases} \end{aligned}$$

Bref, ce morphisme envoie chaque variable gelée sur 1 et chaque variable non-gelée sur elle-même. En particulier, ce morphisme ne crée aucune nouvelle variable amassée. On passe donc d'une algèbre avec $n+m$ variables à une algèbre avec n variables.

Théorème 4.17. Soient $\Sigma = (X, Y, (Q, F))$ et $\Sigma' = (X, U)$.

Alors le morphisme de spécialisation induit une bijection $\Sigma \xrightarrow{\sim} \Sigma'$.

DÉMONSTRATION. Pour obtenir une preuve de ce théorème, on réfère le lecteur à [FZ07].

Exemple 4.18. Soient $(Q, F) = \boxed{2} \longrightarrow 1 \longleftarrow \boxed{3}$ et ρ , le morphisme de spécialisation.

$$\begin{array}{ccc} \rho : \mathbb{Q}(x_1, x_2, x_3) & \longrightarrow & \mathbb{Q}(x_1) \\ x_1 & \longmapsto & x_1 \\ x_2 & \longmapsto & 1 \\ x_3 & \longmapsto & 1 \\ \{x_1, \frac{1+x_2x_3}{x_1}\} & \xrightarrow{\rho} & \{x_1, \frac{2}{x_1}\} \end{array}$$

Le morphisme de spécialisation est le lien explicite entre les algèbres amassées sans coefficients et celles avec coefficients.

Références

- [Cha11] M. Chabot *Frises et récurrences linéaires*, soumis pour publication dans CaMUS.
- [Dou11] G. Douville. *Triangulations, carquois et théorème de Ptolémée*, soumis pour publication dans CaMUS.
- [FZ02] S. Fomin et A. Zelevinsky. *Cluster algebras I : Foundations*. J. Amer. Math. Soc., 15 :497-539, 2002.
- [FZ03] S. Fomin et A. Zelevinsky. *Cluster algebras II : Finite type classification*. Inventiones Mathematicae, 154 :63-121, 2003.
- [FZ07] S. Fomin et A. Zelevinsky. *Cluster algebras IV : Coefficients*. Composition Mathematica, 143(1) :112-164, 2007.
- [Ngu06] B. Nguefack. *Introduction aux Algèbres amassées : Définitions et exemples*, Rapports de recherche de l'université de Sherbrooke, 41 :1-33, 2007.
- [MSW09] G. Musiker, R. Schiffler et L. Williams. *Positivity for cluster algebras from surfaces*, arXiv :0906.0748.

DAVID BOULET-ST-JACQUES, STAGIAIRE, DÉPARTEMENT DE MATHÉMATIQUES, UNIVERSITÉ DE SHERBROOKE, ÉTUDIANT, DÉPARTEMENT DE MATHÉMATIQUES, UQAM
Courriel: david.boulet@hotmail.com

HOMOLOGIE D'UNE SURFACE DE RIEMANN ET PROBLÈMES DE RIEMANN-HILBERT

FÉLIX BARIL BOUDREAU

RÉSUMÉ. Dans cet article nous introduisons d'abord quelques notions fondamentales liées aux surfaces de Riemann, à l'homologie et à l'homologie relative d'une surface. Puis, nous faisons un survol du problème de Riemann-Hilbert appliqué aux revêtements ramifiés de la sphère de Riemann avec tous les points de branchement finis comme dans [Shr08]. Enfin, nous analysons le cas où de tels revêtements possèdent des points de branchement à l'infini et de leurs liens avec les revêtements étudiés dans [Shr08].

1. Approche intuitive du revêtement ramifié

1.1. Notion de coupure

Considérons la fonction complexe

$$(1) \quad f(z) = \sqrt{z} \quad \text{avec} \quad z \in \overline{\mathbb{C}} = \mathbb{C} \cup \{\infty\}.$$

Cette fonction possède deux valeurs en chaque point sauf aux points $z = 0$ et $z = \infty$ où la fonction n'admet qu'une unique valeur. Nous pouvons aussi dire que les deux valeurs coïncident en ces points.

Voyons ce qui se passe dans un voisinage du point $z = 0$. Soit

$$(2) \quad z = re^{i\theta}, \quad 0 \leq \theta \leq 2\pi, \quad r > 0$$

où $z = r$ lorsque $\theta = 0$ ou $\theta = 2\pi$ (voir la figure 1).

Commençons au point $z = r$ avec la valeur \sqrt{r} de la fonction f et faisons faire un tour complet à z autour de $z = 0$. Nous remarquons qu'en revenant au point $z = r$, la fonction change de signe et passe à $-\sqrt{r}$.

Pour le vérifier, fixons d'abord la valeur de \sqrt{r} . Ceci nous donne l'équation suivante

$$(3) \quad f(z) = \sqrt{r}e^{i\theta/2}.$$

Lorsque $\theta = 0$, nous obtenons $f(z) = \sqrt{r}e^0 = \sqrt{r}$.

Cependant, si $\theta = 2\pi$, nous obtenons plutôt $f(z) = \sqrt{r}e^{\pi i} = -\sqrt{r}$.

Ainsi, la fonction $f(z) = \sqrt{z}$ n'est pas correctement définie sur $\overline{\mathbb{C}}$. Afin d'obtenir une fonction univoque sur tout le plan, il faut empêcher l'argument z de la fonction de faire un tour complet autour de $z = 0$ ou de $z = \infty$. Pour

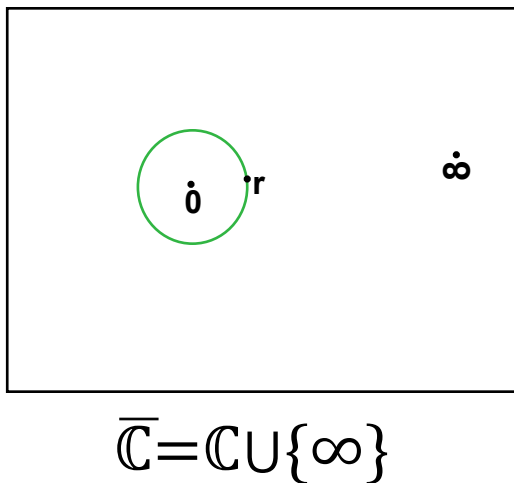
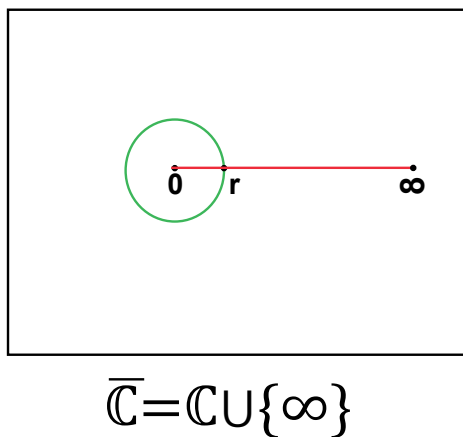


FIGURE 1. Le plan complexe compactifié

FIGURE 2. Une coupure entre 0 et ∞ .

ce faire, nous traçons une courbe dans le plan complexe compactifié reliant les points $z = 0$ et $z = \infty$ et nous coupons ce plan le long de cette courbe. Dans cet article, nous choisissons de tracer la courbe le long de l'axe réel positif (voir la figure 2).

Remarquons maintenant qu'en évaluant la fonction précédente en $z = 1$ avec $\theta = 0$ nous obtenons $f(1) = 1$. Après un tour, donc lorsque $\theta = 2\pi$, la valeur de la fonction devient $f(1) = -1$. Définissons deux branches à la fonction f et utilisons le fait que le plan est coupé entre 0 et ∞ . La première branche, $f_1(z) = \sqrt{z}$, prend la valeur 1 en $z = 1$ et la deuxième branche, $f_2(z) = \sqrt{z}$, est définie telle que $f_2(1) = -1$. Prenons maintenant deux copies de $\overline{\mathbb{C}}$ coupées entre 0 et ∞ que nous nommerons \mathcal{L}_1 et \mathcal{L}_2 (voir la figure 3).

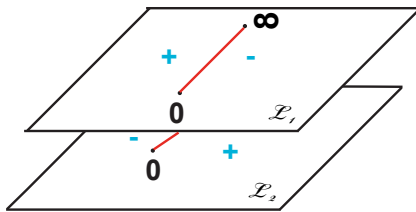


FIGURE 3. Deux copies du plan coupé.

Définissons ensuite la fonction f_1 dans la première des deux copies (\mathcal{L}_1) et f_2 dans l'autre. Nous voulons réussir à coller les deux copies de $\overline{\mathbb{C}}$ ensemble et coller les images des deux branches de la fonction f . L'idée derrière ces manipulations est de construire un domaine sur lequel la fonction \sqrt{z} est univoque.

Tout d'abord, il faut remarquer que chaque copie de $\overline{\mathbb{C}}$ possède deux rives qui sont séparées par une coupure. Chacune de ces rives représente un signe que prend la valeur en 1 de la fonction f sur ce plan. Pour coller les domaines nous procédons ainsi : nous collons la rive de \mathcal{L}_1 où $f_1(1) = 1$ avec la rive de \mathcal{L}_2 où $f_2(1) = 1$. De la même façon, nous collons la rive de \mathcal{L}_1 où $f_1(1) = -1$ avec la rive de \mathcal{L}_2 où $f_2(1) = -1$. Il n'est pas facile de représenter graphiquement un tel collage. Pour mieux apprécier le résultat, il faudrait plonger la surface composée des plans \mathcal{L}_1 et \mathcal{L}_2 dans un espace de dimension supérieure.

Collons maintenant les images des deux branches de la fonction f ensemble. Pour $0 \leq \theta \leq 2\pi$, nous choisissons l'argument de f_1 tel que $0 \leq \arg(f_1(z)) \leq \pi$. Ainsi, l'image de f_1 correspond au demi-plan avec l'argument de f_1 qui varie entre 0 et π . De même, l'image de la fonction f_2 correspond au demi-plan avec l'argument de f_2 variant entre π et 2π . Nous collons ensuite les images des fonctions le long de l'axe réel. L'image de la fonction f correspond alors au plan complexe compactifié en entier.

Nous dirons qu'une surface composée de deux plans que l'on colle de cette façon s'appelle une surface de Riemann de la fonction $f(z) = \sqrt{z}$. La fonction f est univoque en tout point de cette surface car la coupure oblige la fonction à d'abord passer par les plans \mathcal{L}_1 et \mathcal{L}_2 avant de retourner à son point de départ. Le plan complexe compactifié $\overline{\mathbb{C}}$ se nomme la sphère de Riemann. Il est homéomorphe à une sphère par projection stéréographique (Voir la figure 4 et [And05]).

Définition 1 (Genre d'une surface). *Étant donné une surface sans bord ni ponction, le genre est donné par le nombre de trous sur cette surface.*

Par exemple, une sphère est de genre zéro, un tore est de genre un et une sphère à deux anses est de genre deux.

Définition 2 (Surface hyperelliptique). *Nous définissons une surface hyperelliptique C comme*

$$(4) \quad C = \{(\mu, \lambda) | \mu^2 = (\lambda - \lambda_1)(\lambda - \lambda_2) \dots (\lambda - \lambda_{2g+2}), \lambda, \mu \in \mathbb{C}\}$$

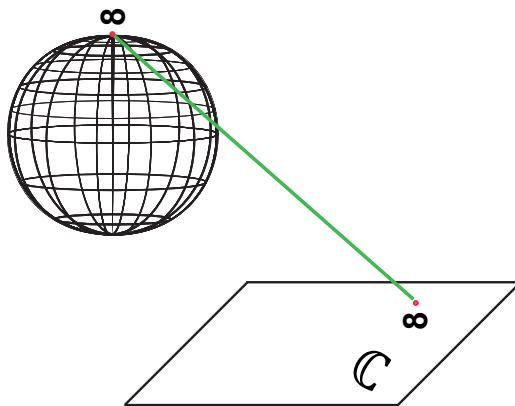
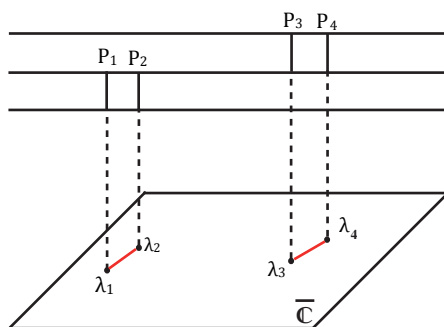


FIGURE 4. Le plan complexe est homéomorphe à la sphère.

FIGURE 5. Le diagramme de Hurwitz $\mathcal{H}_{0;0,0,0}$.

où g représente le genre de la surface. Cependant, le terme *hyperelliptique* s'emploie pour des surfaces avec $g > 1$ tandis qu'une surface avec $g = 1$ sera une *surface elliptique*.

1.2. Le diagramme de Hurwitz et les revêtements ramifiés

Une représentation d'une surface de Riemann est donnée par le diagramme de Hurwitz. Celui-ci est défini comme suit (voir la figure 5). Les lignes horizontales représentent les copies de $\overline{\mathbb{C}}$. Les traits verticaux reliant les deux copies du plan, que nous nommerons maintenant feuillets, représentent les points qui appartiennent à deux feuillets en même temps. Les points sont appelés points de ramification et sont notés P_i avec $i = 1, \dots, n$. Dans le cas de la fonction $f(z) = \sqrt{z}$, les points de ramification sont $z = 0$ et $z = \infty$. Ici, la courbe (4) est l'ensemble des points tels que $(\mu, \lambda) = (0, \lambda_i)$. Enfin, les points de branchement λ_i sont des projections de P_i sur le plan \mathbb{C} . Un ensemble comme celui représenté à la figure 5 est un revêtement ramifié. Ainsi, les points de la surface sont en bijection avec les points du revêtement. Topologiquement, un voisinage d'une coupure liant deux plans est un tuyau liant deux sphères.

Enfin, la surface composée de deux copies de $\overline{\mathbb{C}}$ collées par une coupure est elle-même topologiquement équivalente à une sphère. Ainsi, le collage d'une sphère par une coupure ne change pas le genre de la surface (Voir la figure 6.).

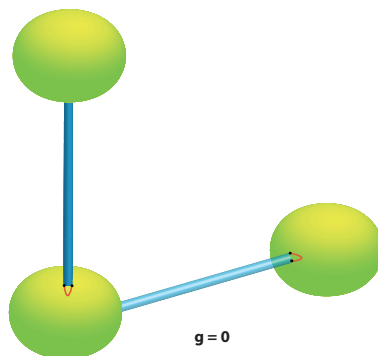


FIGURE 6. Ajouter une nouvelle sphère ne change pas le genre.

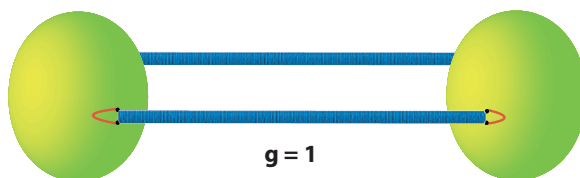


FIGURE 7. Ajouter une paire de tuyaux entre deux sphères augmente le genre de 1.

Cependant, avec deux coupures reliant deux plans, et donc deux tuyaux reliant deux sphères, il n'est pas possible de se ramener à une sphère ; nous obtenons ici un tore. Donc, avec deux coupures le genre augmente de un (voir la figure 7).

Le lien entre le diagramme de Hurwitz et les surfaces est le suivant : les feuillettes sont des sphères et les paires de points de ramification reliés par une coupure sont des tuyaux. La surface de la figure 6 donne une représentation topologique du revêtement ramifié de la figure 5. Le diagramme de Hurwitz correspondant à la surface de la figure 7 a 2 feuillettes et 4 points de ramification.

1.3. Homologie d'une surface

1.3.1. *Groupe d'homologie d'une surface.* Nous donnons d'abord la définition formelle d'homologie et ensuite nous parlerons de la même notion de façon plus intuitive à la fin de cette section.

Il existe en géométrie une notion qui généralise le concept de triangle et que l'on appelle simplexe ou n -simplexe. Nous dirons par exemple qu'un point est un 0-simplexe, qu'un segment de droite est un 1-simplexe, qu'un triangle est un 2-simplexe et qu'un tétraèdre est un 3-simplexe.

De plus, nous définissons un complexe simplicial :

Définition 3 (Complexe simplicial). *Un complexe simplicial est un ensemble fini K de simplexes dans \mathbb{R}^n tel que*

- (1) *si $s \in K$ et $t \subset s$, alors $t \in K$.*
- (2) *si $s, t \in K$ alors $s \cap t$ est soit vide, soit une face de s et de t .*

Ici une face d'un simplexe correspond à un sous-ensemble non-vide de ses sommets. Les faces d'un simplexe sont donc les enveloppes convexes des sous-ensembles de sommets. Nous avons aussi besoin de la notion de suite exacte et celle de p -chaîne.

Définition 4 (Suite exacte). *Une suite*

$$(5) \quad \cdots \rightarrow G_0 \xrightarrow{f_1} G_1 \xrightarrow{f_2} G_2 \xrightarrow{f_3} \cdots$$

de groupes abéliens G_i et d'homomorphismes f_i est dite exacte pour G_1 si $\text{Im} f_1 = \text{Ker} f_2$. De plus, la suite sera dite exacte si elle l'est pour chaque groupe sauf pour le premier et le dernier groupe de la suite.

Définition 5 (p -chaîne). *Une p -chaîne dans K est une combinaison linéaire à coefficients entiers de p -faces du complexe simplicial K .*

Ensuite, nous disons que la dimension d'un complexe simplicial est n si n est la dimension maximale que peut avoir la face d'un simplexe pour tout simplexe dans K . S'il n'existe pas de tel n , alors le complexe simplicial est de dimension infinie.

Soient un complexe simplicial orienté K de dimension n , α_p le nombre de p -simplexes de K et pour $0 \leq p \leq n$, $\sigma_p^1, \dots, \sigma_p^{\alpha_p}$ les p -simplexes orientés de K . Pour un tel p , le p -ième groupe de chaîne de K , que nous notons $C_p(K)$, est un groupe abélien libre engendré par l'ensemble $\{\sigma_p^1, \dots, \sigma_p^{\alpha_p}\}$ pour l'addition formelle de simplexes. De plus, un élément de $C_p(K)$ est une combinaison linéaire de simplexes de la forme $\lambda_1 \sigma_p^1 + \dots + \lambda_{\alpha_p} \sigma_p^{\alpha_p}$ où $\lambda_i \in \mathbb{Z}$. Enfin, nous définissons le bord d'un simplexe $\sigma = (s_0 \cdots s_n)$, avec s_i les sommets du simplexe, comme la $(p-1)$ -chaîne $\partial \sigma = \partial_p \sigma = \sum_{i=0}^p (-1)^i (s_0 \cdots s_{i-1} s_{i+1} \cdots s_p)$.

Soit

$$(6) \quad C_{p+1} \xrightarrow{\partial_{p+1}} C_p(K) \xrightarrow{\partial_p} C_{p-1}(K).$$

Tout d'abord nous remarquons que les éléments de $\text{Ker} \partial_p$ sont des p -cycles (Un p -cycle étant une p -chaîne où le premier et le dernier sommets sont égaux.). Par ailleurs, $\text{Ker} \partial_p$ est un groupe abélien libre. De plus, $\text{Im} \partial_{p+1}$ est également un groupe abélien libre et ses éléments sont des p -bords. Comme tout bord est un cycle, il est possible de montrer que $\text{Im} \partial_{p+1} \subset \text{Ker} \partial_p$ et donc il existe un groupe quotient, $H_p(K) = \text{Ker} \partial_p / \text{Im} \partial_{p+1}$. Ce groupe quotient se nomme le p -ième groupe d'homologie de K et permet de savoir combien de p -cycles qui ne sont pas des bords le complexe simplicial K possède. Cela revient à connaître le nombre de cycles fermés non-triviaux différents de dimension p dans K . Il est possible d'obtenir plus de détails sur cette théorie en consultant [Gib77].

Dans cet article nous ne considérons que des surfaces et nous n'aurons besoin que des 0-simplexes, des 1-simplexes et des 2-simplexes. C'est pourquoi nous allons uniquement nous intéresser au premier groupe d'homologie de la surface. Dans ces conditions, les éléments du groupe d'homologie de la surface seront les classes d'équivalence des cycles fermés non-triviaux différents que l'on peut tracer sur la surface.

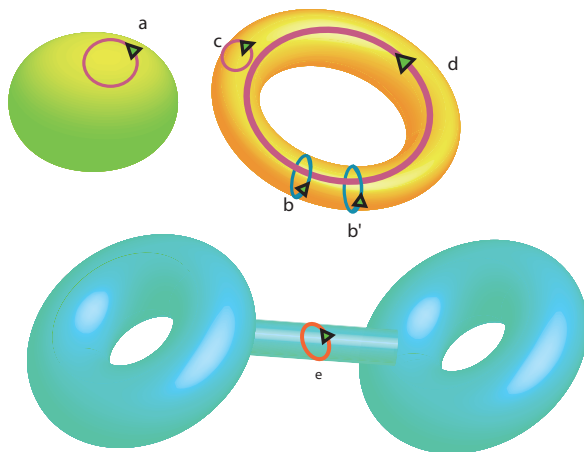


FIGURE 8. Les groupes d'homologie de la sphère, du tore et de la sphère à deux anses

Voyons comment trouver le premier groupe d'homologie des trois surfaces suivantes : la sphère, le tore et la sphère à deux anses. Soient a , b , b' , c et d des 1-cycles (voir la figure 8). Nous remarquons que a et c bordent une région, tandis que b , b' et d n'en bordent aucune. Comme la courbe simple fermée b ne borde pas de région, cela indique la présence d'un trou dans le tore. La même déduction est vraie pour d .

Nous avons remarqué que a et c délimitent une région, ceux-ci sont dits homologues à 0 et nous les notons $a \sim 0$ et $c \sim 0$. De même, nous constatons que b et $-b'$ ensemble définissent un tuyau et donc une région. Cela veut dire que $b - b' \sim 0$ et donc $b \sim b'$. Ainsi, b et b' tournent autour du même trou du tore. De plus, a et c peuvent être rétrécis à un point. Enfin, e borde une région (la moitié de la surface), donc $e \sim 0$.

Il faut d'abord remarquer que nous pouvons formellement additionner les cycles sur une surface (comme nous avons fait ci-dessus en écrivant $b - b'$). Nous obtenons alors un groupe de cycles pour l'addition. Les bords forment un sous-groupe de ce groupe. Le groupe quotient cycles/bords se nomme le premier groupe d'homologie. Les éléments de ce groupe sont des cycles qui ne délimitent aucune région de la surface et donc, le nombre de générateurs de ce groupe est deux fois le nombre de trous de l'objet. De plus, b et d sont des éléments du premier groupe d'homologie du tore. Enfin, la sphère ne possède aucun générateur tandis que le tore en a deux. Ainsi, leur premier groupe d'homologie est respectivement 0 et \mathbb{Z}^2 .

1.3.2. Groupe d'homologie relative d'une surface. Il est aussi possible de trouver l'homologie relative d'une surface. Soit S un sous-ensemble d'une surface X . Identifions tous les points de ce sous-ensemble à un point. Cette manipulation transforme la surface. L'homologie de cette surface transformée est appelée l'homologie de la surface originale X relativement au sous-ensemble S et se note $H_1(X, S)$.

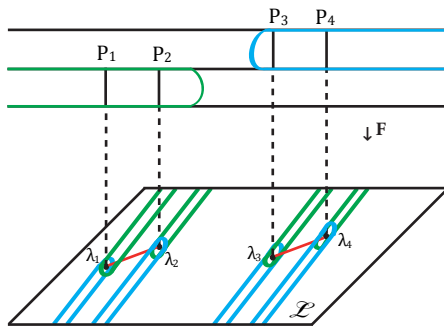


FIGURE 9. Deux systèmes de contours.

2. Problème de Riemann-Hilbert

Cette section donne une brève description du problème de Riemann-Hilbert. Il est possible d'obtenir plus de détails sur ce problème en consultant [Shr08].

Définissons d'abord deux ensembles de contours sur une surface de Riemann X . Considérons aussi une fonction méromorphe F telle que $F : X \rightarrow \overline{\mathbb{C}}$ et le diagramme de Hurwitz associé à F . Le premier système de contours sera la base des contours de gauche \mathcal{C}_j^l et le second système sera la base des contours de droite \mathcal{C}_j^r . Au voisinage de l'infini, les contours de droite sont parallèles à la droite d'argument $3\pi/2 - \phi$. Le contour \mathcal{C}_j^r encercle le point de branchement, λ_i , dans $\overline{\mathbb{C}}$ et revient vers l'infini en restant parallèle à la droite. De même, les contours de gauche resteront parallèles à la droite d'argument $\pi/2 - \phi$ au voisinage de l'infini et encerclent aussi les points de branchement avant de revenir vers l'infini tout en restant parallèle à cette dernière droite. Les contours de gauche sont toujours dans la direction opposée de celle des contours de droite (Voir la figure 9.).

Nous pouvons ensuite lever les contours vers la surface à l'aide du diagramme de Hurwitz. Cette opération est faite de sorte que F projette les contours sur la base du revêtement. Sur le diagramme de Hurwitz, lorsque nous encerclons un point de ramification, nous revenons à l'infini mais sur un autre feuillet (Voir la figure 9.).

Les deux systèmes de contours forment deux ensembles générateurs dans le groupe d'homologie de la surface X relativement à un certain sous-ensemble $Y \subset X$.

La façon dont les systèmes de contours de « gauche » et de « droite » sont définis et comment l'ensemble Y est construit sont décrits en détail dans [Shr08]. Dans cet article, l'auteure cherche à construire deux fonctions à valeurs matricielles Ψ_l et Ψ_r telles que chaque fonction a de bonnes propriétés dans un des deux demi-plans du plan complexe. Les deux demi-plans sont séparés par une droite l qui forme un angle ϕ avec la droite réelle. Les valeurs des deux fonctions sur la droite l sont reliées par multiplication d'une matrice à valeurs constantes que appelée matrice de Stokes. La relation entre ces matrices est donnée par :

$$(7) \quad \Psi_l(z) = \Psi_r(z)S \quad \text{si} \quad z \in l.$$

Ensemble, les deux fonctions à valeurs matricielles peuvent être vues comme une seule fonction dans $\overline{\mathbb{C}}$ qui possède une discontinuité (un saut) de type (7) sur la droite l . Une telle fonction donne une solution au problème de Riemann-Hilbert qui est traité dans [Shr08]. De plus, il a été montré dans [Shr08] que les deux fonctions à valeurs matricielles Ψ_l et Ψ_r peuvent être exprimées comme les intégrales de certaines différentielles définies sur la surface de Riemann correspondante par rapport aux deux systèmes de contours définis plus tôt, les C_j^l et les C_j^r , où $j = 1, \dots, n$.

Il s'avère que les expressions pour Ψ_l et Ψ_r ne diffèrent que par leurs contours d'intégration, c'est-à-dire que les vecteurs des différentielles intégrés sur les contours C_1^r, \dots, C_n^r donnent les colonnes de la matrice Ψ_r et ces mêmes vecteurs intégrés sur les contours C_1^l, \dots, C_n^l donnent les colonnes de la matrice Ψ_l . Ainsi, la matrice de Stokes S n'est qu'une matrice de transformation du système des contours de « droite », les C_1^r, \dots, C_n^r , vers le système des contours de « gauche », les C_1^l, \dots, C_n^l .

Donc, calculer la matrice de Stokes d'un problème du Riemann-Hilbert donné revient à calculer la transformation entre les deux ensembles générateurs de $H_1(X, Y)$; de là l'importance de calculer une telle transformation.

3. Matrice de Stokes avec infini non branché

Dans [Shr08], les transformations entre les deux systèmes de contours ont été trouvées pour des revêtements ramifiés où seuls les points finis étaient des points de ramification. Dans cette section, nous ne faisons que rappeler les relations existant entre les systèmes de contours pour certains revêtements sans entrer dans les détails. Ces relations pourront être comparées à celles obtenues dans la section suivante où l'infini est aussi un point de ramification. La notation utilisée pour représenter un diagramme de Hurwitz est $\mathcal{H}_{g;p_1, \dots, p_i, \dots, p_n}$ où g est le genre de la surface et p_i , $i = 1, \dots, n$, est l'ordre de ramification du ou des points à l'infini. C'est-à-dire le nombre de feuillets -1 que le i -ème point de branchement à l'infini relie.

Pour le revêtement $\mathcal{H}_{0;0,0}$, les contours sont donnés par les relations suivantes :

$$\begin{aligned} C_1^l &= C_1^r - 2C_2^r \\ C_2^l &= C_2^r \end{aligned}$$

et la matrice de Stokes est

$$S^{0;0,0} = \begin{pmatrix} 1 & 0 \\ -2 & 1 \end{pmatrix}.$$

Pour le revêtement $\mathcal{H}_{0;0,0,0}$ de la figure 9 ou 5, les relations sont les suivantes :

$$C_1^l = C_1^r - 2C_2^r - C_3^r + C_4^r$$

$$\begin{aligned}\hat{\mathcal{C}}_2^l &= \hat{\mathcal{C}}_2^r + \hat{\mathcal{C}}_3^r - \hat{\mathcal{C}}_4^r \\ \hat{\mathcal{C}}_3^l &= \hat{\mathcal{C}}_3^r - 2\hat{\mathcal{C}}_4^r \\ \hat{\mathcal{C}}_4^l &= \hat{\mathcal{C}}_4^r\end{aligned}$$

et la matrice de Stokes associée est :

$$\begin{aligned}\mathbf{S}^{0;0,0,0} &= \begin{pmatrix} 1 & 0 & 0 & 0 \\ -2 & 1 & 0 & 0 \\ -1 & 1 & 1 & 0 \\ 1 & -1 & -2 & 1 \end{pmatrix} \\ &= \begin{pmatrix} \mathbf{S}^{0;0,0} & \mathbf{0} \\ \mathbf{A} & \mathbf{S}^{0;0,0} \end{pmatrix}\end{aligned}$$

où

$$\mathbf{A} = \begin{pmatrix} -1 & 1 \\ 1 & -1 \end{pmatrix}.$$

Enfin, pour le revêtement $\mathcal{H}_{1;0,0}$ les relations associées sont :

$$\begin{aligned}\mathcal{C}_1^l &= \mathcal{C}_1^r - 2\mathcal{C}_2^r + 2\mathcal{C}_3^r - 2\mathcal{C}_4^r \\ \mathcal{C}_2^l &= \mathcal{C}_2^r - 2\mathcal{C}_3^r + 2\mathcal{C}_4^r \\ \mathcal{C}_3^l &= \mathcal{C}_3^r - 2\mathcal{C}_4^r \\ \mathcal{C}_4^l &= \mathcal{C}_4^r\end{aligned}$$

De plus, la matrice de Stokes est :

$$\mathbf{S}^{1;0,0} = \begin{pmatrix} \mathbf{S}^{0;0,0} & \mathbf{0} \\ -2\mathbf{A} & \mathbf{S}^{0;0,0} \end{pmatrix}.$$

4. Matrice de Stokes avec infini branché

Dans [Shr08], tous les revêtements étudiés n'ont pas de point de branchement à l'infini. Dans cette section nous nous intéressons aux mêmes revêtements que précédemment, mais en déplaçant le point le plus à droite du diagramme de Hurwitz vers l'infini afin que celui-ci soit branché. Dans ce cas, nous identifions les points à l'infini de deux feuillets adjacents au même point.

Remarque 1. Rappelons que les contours $\{\mathcal{C}_j^r\}$ et $\{\mathcal{C}_j^l\}$ sont des éléments de l'homologie relative $H_1(X, Y)$ pour un certain $Y \subset X$. La structure de Y et donc de $H_1(X, Y)$, est telle que la direction d'approche de l'infini par $\{\mathcal{C}_j^r\}$ peut être modifiée dans l'intervalle $(\frac{3\pi}{2} - \phi, \frac{\pi}{2} - \phi)$ sans changer la classe de \mathcal{C}_j^r dans $H_1(X, Y)$ (Voir [Shr08]). C'est-à-dire qu'une rotation dans le sens horaire de \mathcal{C}_j^r dans un voisinage de ∞ ne change pas de classe de contour. Une déformation quelconque loin de ∞ est également permise.

Le diagramme de Hurwitz correspondant à $\mathcal{H}_{0,1}$ est donné à la figure 10. En déformant les contours de façon continue, nous trouvons que $\mathcal{C}_1^l = \mathcal{C}_1^r$. Ainsi, la matrice de Stokes de ce revêtement est donnée par

$$\mathbf{S}^{0;1} = 1.$$

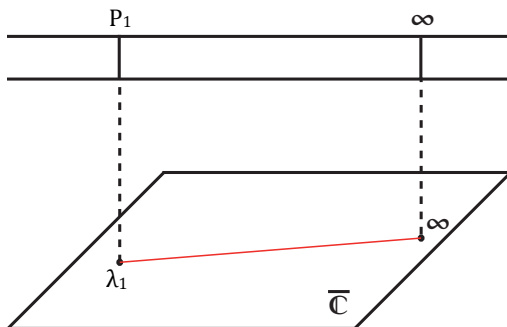


FIGURE 10. Diagramme de Hurwitz du revêtement $\mathcal{H}_{0;1}$

Nous montrerons une méthode pour trouver les contours à partir du revêtement $\mathcal{H}_{1;1}$. Les relations de contours dans les revêtement $\mathcal{H}_{0;1}$ et $\mathcal{H}_{0;0,1}$ se trouvent de la même façon. Le diagramme de Hurwitz correspondant au revêtement $\mathcal{H}_{1;1}$ est donné à la figure 11.

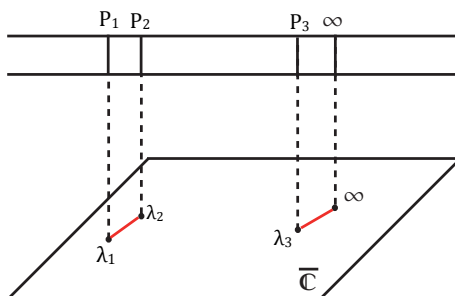


FIGURE 11. Diagramme de Hurwitz du revêtement $\mathcal{H}_{1;1}$

Il est souvent plus commode de faire des manipulations sur les contours en choisissant la représentation de la figure 12. Le segment uni indique que le contour se trouve sur le premier feuillet tandis que le segment pointillé qu'il est plutôt sur le deuxième.

Débutons avec les contours du point P_3 que nous pouvons voir à la figure 13. Ici, nous n'avons pas besoin de tenir compte des autres contours parce que la rotation du contour de droite se fait dans le sens horaire vers le contour de gauche. Celui-ci ne risque donc, pas de rencontrer d'autre point de branchement. Nous gardons également le contour de gauche à droite de l'image afin de nous rappeler l'objectif à atteindre.

Nous pouvons tirer le contour pointillé et lui faire traverser la coupure. En la traversant, le contour change de feuillet (Voir la figure 14). Nous déformons ensuite le contour dans le voisinage de P_3 pour obtenir la figure 15. Puis, nous

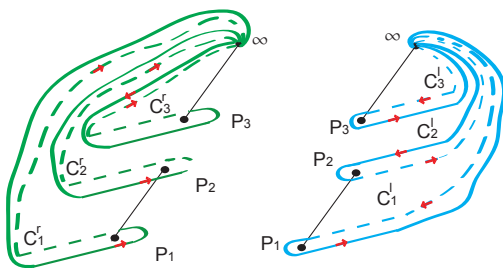
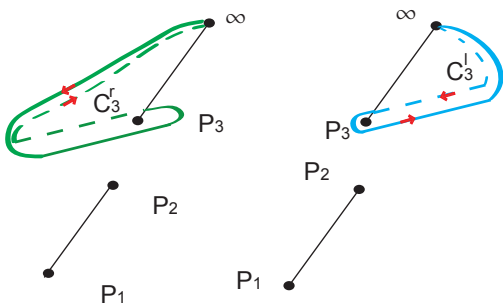
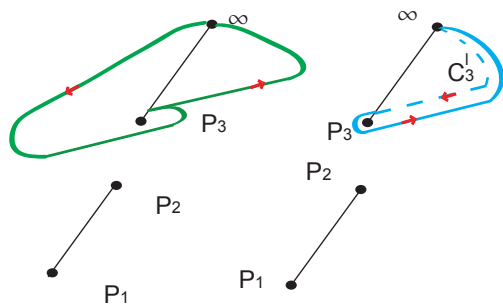
FIGURE 12. Représentation plus pratique de $\mathcal{H}_{1,1}$ FIGURE 13. Relations de contours pour le point P_3 

FIGURE 14. En traversant la coupure, cette partie du contour change de feuillet.

tirons l'autre partie du contour pour la faire traverser la coupure. Ceci nous donne la figure 16 et nous remarquons que les deux contours sont identiques. Cela veut dire que $\mathcal{C}_3^r = \mathcal{C}_3^l$ car nous n'avons que déformé \mathcal{C}_3^r de la façon permise dans la Remarque 1.

Dans la section précédente nous avons vu que pour le point P_2 , la relation entre les contours était donnée par $\mathcal{C}_2^l = \mathcal{C}_2^r - 2\mathcal{C}_3^r + 2\mathcal{C}_4^r$. Il serait tout à fait raisonnable de supposer que cette relation est conservée mais en supprimant le contour de P_4 puisque c'est le point ∞ qui le remplace. Dans l'image suivante (la figure 17), nous avons tracé le contour \mathcal{C}_2^r et nous lui avons ajouté un premier $-\mathcal{C}_3^r$ puisque le contour est de signe opposé dans cette relation. Voyons si nous avons suffisamment d'éléments pour obtenir le contour de gauche. Il est important de

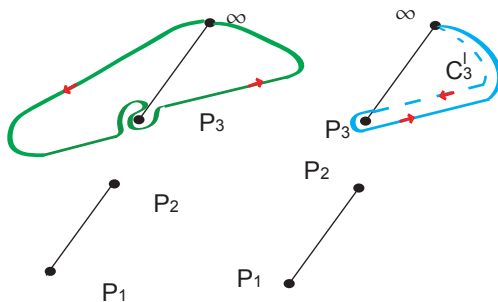


FIGURE 15. Nous déformons le contour comme indiqué à la Remarque 1.

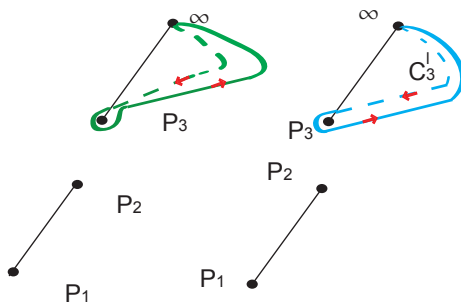


FIGURE 16. Cette dernière manipulation montre que $\mathcal{C}_3^r = \mathcal{C}_3^l$.

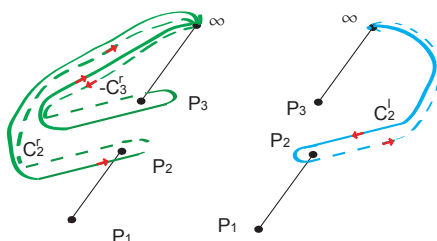


FIGURE 17. Nous débutons avec les contours \mathcal{C}_2^r et $-\mathcal{C}_3^r$.

mentionner qu'ici il faut tenir compte du point de branchement P_3 car la rotation du contour de droite au point P_2 va rencontrer cet autre point.

Les deux prochaines étapes se passeront de commentaires puisqu'elles correspondent exactement aux mêmes manipulations réalisées pour P_3 .

Visiblement, ces deux contours ne sont pas suffisants pour obtenir l'image de droite. Ajoutons un second contour de la forme $-\mathcal{C}_3^r$ (en mauve dans la figure 18). Nous pouvons maintenant déplacer certaines parties des contours près de la coupure dans un voisinage de P_3 pour obtenir la figure 19.

Le contour \mathcal{C}_2^r est équivalent à celui que nous obtenons à la figure 20 puisque ce contour approche et revient vers l'infini dans les mêmes directions que \mathcal{C}_2^r . Il ne s'agit encore une fois qu'une transformation près des points de branchement.

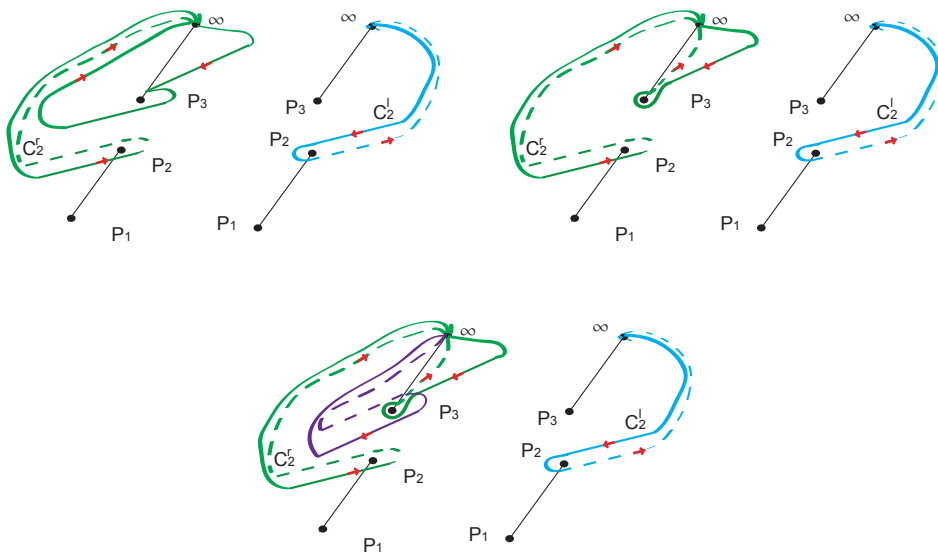
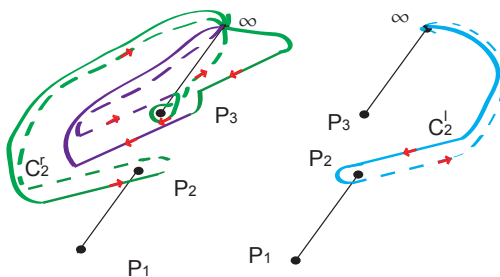
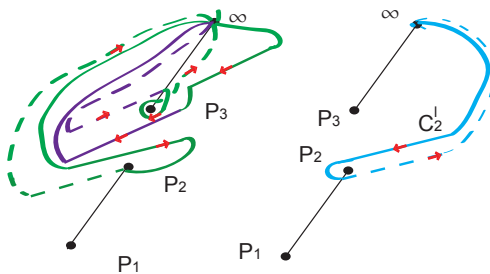
FIGURE 18. Ajoutons un second contour de la forme $-\mathcal{C}_3^r$.

FIGURE 19. Quelques manipulations près de la coupure

FIGURE 20. Cette manipulation donne un contour équivalent à \mathcal{C}_2^r .

Ensuite, nous tirons la partie en vert et en mauve du contour sur le premier feuillet (Voir la figure 21). Nous procédons de même pour la partie du contour sur le deuxième feuillet (Voir la figure 22.). Déplaçons légèrement les contours afin de clarifier la situation (Voir la figure 23.).

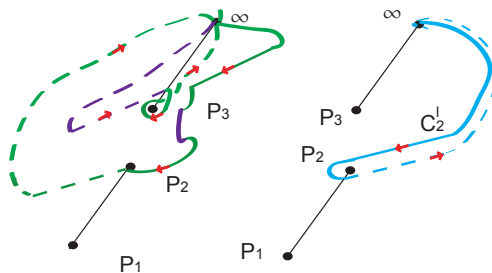


FIGURE 21. Nous déplaçons la partie du contour sur le premier feuillet vers la droite de la figure.

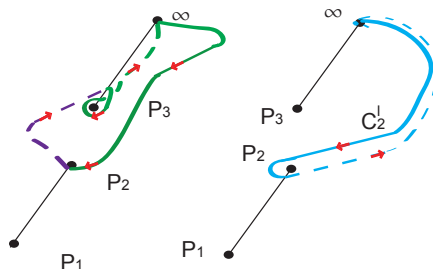


FIGURE 22. Nous déplaçons la partie du contour sur le deuxième feuillet vers la droite de la figure.

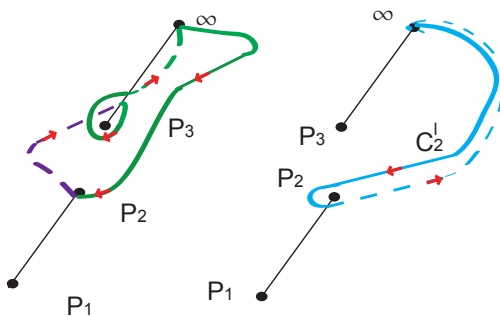


FIGURE 23. Léger déplacement du contour près de la coupure

Nous pouvons encore une fois déformer la partie du contour sur le deuxième feuillet. L'idée est d'obtenir un contour équivalent au nul autour de P_3 que nous pourrions enlever (Voir la figure 24.). Un tour complet autour d'un point de ramification simple est un élément nul dans l'homologie.

En reliant différemment certaines parties du contour, nous avons alors notre contour équivalent au contour nul et nous pourrions le supprimer à la prochaine étape (Voir la figure 25.).

Le contour ainsi obtenu est équivalent au contour de l'illustration de droite puisque la direction de départ et d'arrivée vers l'infini est la même et le contour se fait au même point de branchement (Voir la figure 26.).

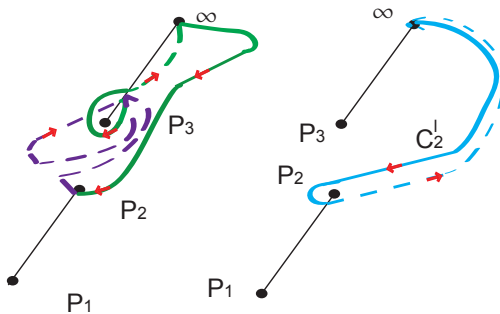


FIGURE 24. L'idée est d'obtenir un contour équivalent au nul autour de P_3 .

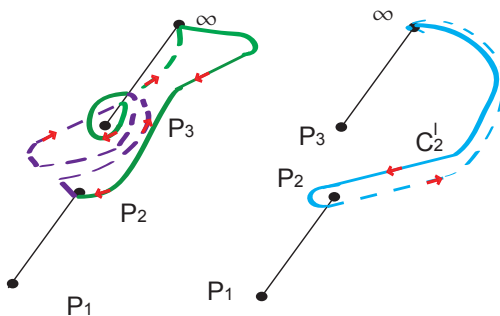


FIGURE 25. Un contour équivalent au nul autour de P_3 est obtenu.

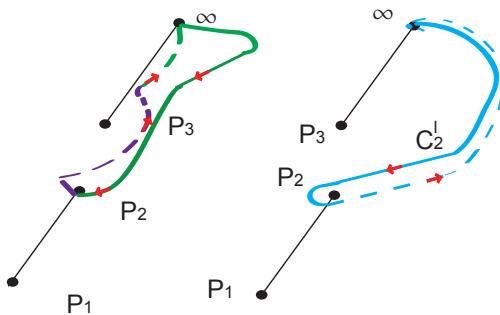


FIGURE 26. Le contour obtenu est équivalent au contour de l'illustration de droite.

En suivant la même idée, nous obtenons la relation de contours associée au point P_1 . Les relations obtenues sont alors :

$$\mathcal{C}_1^l = \mathcal{C}_1^r - 2\mathcal{C}_2^r + 2\mathcal{C}_3^r$$

$$\mathcal{C}_2^l = \mathcal{C}_2^r - 2\mathcal{C}_3^r$$

$$\hat{\mathcal{C}}_3^l = \hat{\mathcal{C}}_3^r.$$

La matrice de Stokes correspondante est

$$\mathbf{S}^{1;1} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ 2 & -2 & 1 \end{pmatrix}.$$

Enfin, le diagramme de Hurwitz du revêtement $\mathcal{H}_{0;0,1}$ est presque identique à $\mathcal{H}_{0;0,0,0}$ de la figure 5, sauf pour le point P_4 qui est remplacé par le point à l'infini. Les relations de contours de ce revêtement sont

$$\mathcal{C}_1^l = \mathcal{C}_1^r - 2\mathcal{C}_2^r - \mathcal{C}_3^r$$

$$\mathcal{C}_2^l = \mathcal{C}_2^r + \mathcal{C}_3^r$$

$$\mathcal{C}_3^l = \mathcal{C}_3^r$$

et la matrice de Stokes associée est :

$$\mathbf{S}^{0;0,0,0} = \begin{pmatrix} 1 & 0 & 0 \\ -2 & 1 & 0 \\ -1 & 1 & 1 \end{pmatrix}.$$

Nous constatons que le fait d'avoir l'infini comme point de branchement revient à supprimer la dernière ligne et la dernière colonne de la matrice de Stokes obtenue avec tous les points de branchement finis. Ce résultat peut être déduit directement de la théorie des revêtements ramifiés en prenant la limite de la fonction méromorphe F .

5. Méthode des serpents et échelles

Parfois, il peut s'avérer ardu de trouver les relations entre les contours. Nous avons développé une méthode mnémonique afin de faciliter le calcul de la matrice de Stokes. Voici les règles à suivre. Soit X , une surface de Riemann et soit son diagramme de Hurwitz associé à une fonction $F : X \rightarrow \overline{\mathbb{C}}$. Tout tracé de contour débute à la droite du diagramme de Hurwitz. Si le contour de gauche \mathcal{C}_k^l dont on cherche la relation en termes de contours de droite est d'indice impair, il faut commencer sur le feuillet du bas du point P_k correspondant. Si le contour de gauche est d'indice pair, il faut commencer sur le feuillet du haut de ce contour. Une fois le point de départ identifié, nous nous déplaçons vers la gauche du diagramme de Hurwitz en respectant la règle suivante : à chaque fois que l'on rencontre le bas d'un point de branchement, il faut passer au feuillet du haut et écrire \mathcal{C}_i^r . Au contraire, si l'on se retrouve sur le feuillet du haut, il faut passer au feuillet du bas et écrire $-\mathcal{C}_i^r$, où i est l'indice du point de branchement rencontré. Lorsque nous avons fait cette opération pour le point de branchement P_k , nous poursuivons le tracé en se déplaçant vers la droite du diagramme de Hurwitz et en appliquant la même règle d'écriture des contours que plus haut. Le premier point de branchement sur lequel nous effectuons cette opération est le dernier rencontré avant P_k . Une fois rendu à l'extrémité droite du diagramme, nous faisons la somme des éléments inscrits. Cette somme correspond à la combinaison linéaire de contours de droite permettant d'exprimer \mathcal{C}_k^l . Nous répétons le processus pour chaque contour de gauche, jusqu'à ce que nous ayons toutes les relations. Comme chaque relation correspond à la k -ième colonne de la matrice de Stokes, nous obtenons la matrice au complet une fois que tous les points de branchement auront été analysés.

Remerciements. L'auteur tient à remercier Vasilisa Shramchenko pour son aide et ses judicieux conseils tout au long de la réalisation du projet et de la rédaction de l'article. L'auteur tient également à remercier Caroline Hébert pour son soutien moral et pour son aide lors de l'utilisation du logiciel *Adobe Illustrator* dont sont issues les illustrations de l'article. Ce premier article lui est dédié.

Références

- [And05] James W. Anderson. *Hyperbolic geometry*. Springer, London, 2005.
- [Gib77] Peter John Giblin. *Graphs, Surfaces and Homology*. Chapman and Hall, London, 1977.
- [Shr08] Vasilisa Shramchenko. Riemann-hilbert problem associated to frobenius manifold structures on hurwitz spaces : irregular singularity. *Duke Mathematical Journal*, 144(1) :1-52, 2008.

FÉLIX BARIL BOUDREAU, DÉPARTEMENT DE MATHÉMATIQUES, UNIVERSITÉ DE SHERBROOKE

Courriel: Felix.Baril.Boudreau@USherbrooke.ca